

TREBALL FI DE GRAU

Grau en Enginyeria Química

APLICACIÓ INFORMÀTICA PER A LA CONSTRUCCIÓ DE MATRIUS POLIMÈRIQUES AMORFES D'ALTA DENSITAT



Memòria i Annexos

Autor:	Montse Català
Director:	Joan Torras
Convocatòria:	Gener 2018

Resum

L'objectiu d'aquest projecte ha sigut iniciar el desenvolupament d'un programa que sigui capaç de construir matrius polimèriques amorfes d'alta densitat.

Les matrius polimèriques poden estar compostes de cadenes molt llargues que poden ser d'un únic tipus de monòmer o de diferents tipus. Poden contenir cadenes de copolímer trifàsic, cadenes ordenades aleatòriament, estrelles, anells, barreges compostes de diferents longituds de cadena, diferents monòmers de ramificació...; per això degut a la seva estructura interna, són molt complexes de comprendre i preveure el seu creixement o comportament a escala molecular o realitzar simulacions realistes.

Per aquest motiu, i prenent com a referència els estudis iniciats en els mètodes de simulació SuSi on ja es presentava una nova estratègia computacional per generar estructures atòmiques representatives de polímers densos; s'ha desenvolupat un nou programa en Python.

A partir d'uns fitxers d'entrada que inclouen la composició del sistema que s'ha de trobar i d'altres dades necessàries; el programa és capaç d'anar enllaçant els diferents monòmers seguint un ordre i lògica establert obtenint com a resultat un fitxer en format PDB. Aquest fitxer conté el sistema que pot ser emprat per altres programes en la realització de simulacions atomístiques.

Resumen

El objetivo de este proyecto ha sido iniciar el desarrollo de un programa capaz de construir matrices poliméricas amorfas de alta densidad.

Las matrices poliméricas pueden estar compuestas de cadenas muy largas que pueden ser de un único tipo de monómero o diferentes tipos. Pueden contener cadenas de copolímero trifásico, cadenas ordenadas aleatoriamente, estrellas, anillos, mezclas compuestas de diferentes longitudes de cadena, diferentes monómeros de ramificación ...; por ello debido a su estructura interna, son muy complejas de comprender y prever su crecimiento o comportamiento a escala molecular o realizar simulaciones realistas.

Por este motivo, y tomando como referencia los estudios iniciados en los métodos de simulación Susi donde ya se presentaba una nueva estrategia computacional para generar estructuras atómicas representativas de polímeros densos; Se ha desarrollado un nuevo programa en Python.

A partir de unos ficheros de entrada que incluyen la composición del sistema que se tiene que encontrar y de otros datos necesarios; el programa es capaz de ir enlazando los diferentes monómeros siguiendo un orden y lógica establecida, obteniendo como resultado un fichero en formato PDB. Este fichero contiene el sistema que puede ser utilizado por otros programas en la realización de simulaciones atomísticas.

Abstract

The objective of this project has been to initiate the development of a program capable to build one amorphous polymer matrices of high density.

The polymer matrices can be composed of very long chains, which may be of a single monomer type or different type. They can also contain three-phase copolymer chains, randomly ordered chains, stars, rings, mixtures composed of different chain lengths, different branching monomers ...; Therefore, due to their internal structure, they are very complex to understand and foresee their growth or behavior on a molecular scale or to carry out realistic simulations.

For this reason, and taking as a reference the studies initiated in the Susi simulation methods where a new computational strategy was already presented to generate representative atomic structures of dense polymers; A new computer program has been developed with Python.

Through some input files that contain the composition a system that has to be found and other necessary data; the program is capable to link the different monomers following an order and logic established, obtaining as result an output file in PDB format. This file contains the system that can be used with other programs to perform atomistic simulations.

Agraïments

Vull agrair a Joan Torras i Guillem Ruano la seva col·laboració i ajuda durant la realització del projecte. Així com també agraeixo el suport de la meva família i amics durant aquests llargs mesos.

Glossari

Resum dels principals símbols, signes, abreviatures, acrònims i termes del projecte que poden no ser compresos fàcilment i ràpidament:

Algoritme de programació: Es una sèrie de passos organitzats que descriu el procés que s'ha de seguir, per donar solució a un problema específic.

Array: És un tipus de dada estructurada que permet emmagatzemar un conjunt de dades en la memòria del dispositiu, totes elles del mateix tipus i relacionades.

Binding: En programació, un binding és una adaptació d'una biblioteca per ser utilitzada en un llenguatge de programació diferent d'aquell al que ha estat escrita

Biblioteca (Library): Col·lecció o conjunt de subprogrames utilitzats per desenvolupar software que ofereix funcionalitats molt específiques a l'usuari, en forma de subrutines que funcionen de forma autònoma.

Classe: És una plantilla per la creació d'objectes de dades segons un model predefinit.

Float: És una dada que corresponen als números reals

Framework: En el desenvolupament de software, un framework o infraestructura digital, es una estructura conceptual i tecnològica de suport definit, normalment amb artefactes o mòduls concrets de software, que poden servir de base per l'organització i desenvolupament de software.

Integer: És una dada que representa els números enters; poden prendre valors positius i negatius, sense part decimal.

Python: és un llenguatge de programació d'alt nivell i propòsit general molt utilitzat. La seva filosofia de disseny busca llegibilitat en el codi i la seva sintaxi permet als programadors expressar conceptes en menys línies de codi del que seria possible en llenguatges com C. També proveeix estructures per permetre programes més entenedors tant a petita com a gran escala

Script: es un arxiu que conté un guió amb el conjunt d'ordres i passos per el processament de dades, escrits en llenguatge de codi de programació i emmagatzemats en un arxiu de text pla.

String: es un tipus de dades que conté una seqüència ordenada de caràcters utilitzada habitualment per l'emmagatzemat i processat de paraules, frases o qualsevol altre successió de caràcters.

Debugger: s'utilitza per provar i depurar altres programes. El codi que s'ha d'examinar alternativament es pot executar en un simulador de conjunts d'instruccions (ISS), una tècnica que permet una gran potència en la seva capacitat d'aturar quan es troben condicions específiques, però que normalment seran una mica més lent que executar el codi directament en el cas apropiat (o el mateix). Alguns depuradores ofereixen dos modes de funcionament, simulació total o parcial, per limitar aquest impacte.

Break point: En el desenvolupament, un break point és un lloc de detenció o pausa intencionat en un programa, establert per depuració. De vegades, simplement es fa referència a una pausa.

Índex

RESUM	I
RESUMEN	II
ABSTRACT	III
AGRAÏMENTS	IV
GLOSSARI	V
1. INTRODUCCIÓ	13
1.1. Objectius del treball	13
1.2. Abast del treball	13
1.2.1. Origen dels polímers	14
1.2.2. Classificació dels polímers	14
1.2.2.1. Termoplàstics	15
1.2.2.2. Termoestables	15
1.2.3. Estructura dels polímers	16
1.2.3.1. Estructura química dels polímers	16
1.2.3.2. Estructura física dels polímers i cristal·lització	18
1.2.4. Matrius polimèriques i simulacions	21
2. ESTAT DE L'ART	23
2.1. Antecedents	23
2.2. Nou mètode de simulació d'estructura (SuSi)	23
2.3. Estudi sobre el rendiment	24
2.4. Estratègia teòrica per proporcionar models atòmics de polímers comb-like	26
2.5. influència del mètode de relaxació	29
3. ANÀLISIS	31
3.1. Necessitat del programa	31
3.2. Model conceptual	31
3.3. Classes principals	33
3.3.1. SYSTEM	33
3.3.2. MOLECULE	33
3.3.3. RESIDUE	34
3.3.4. ATOM	34
3.4. Classes d'entrada i sortida	35

3.4.1.	Fitxers d'entrada	35
3.4.1.1.	Fitxers de càrrega	36
3.4.1.2.	Fitxers PREPI	37
3.4.2.	Fitxers de sortida	39
3.4.2.1.	Fitxer format PDB	39
3.4.2.2.	Missatges	42
3.4.3.	Contingut de les classes	42
3.4.3.1.	FILES	42
3.4.3.2.	FILEPDB	42
3.4.3.3.	FILEPREPI	42
3.4.3.4.	MESSAGES	43
3.5.	Classes de processament	44
3.5.1.	BUILDER	44
3.5.2.	INT2CAR_M	47
4.	IMPLEMENTACIÓ	48
4.1.	BUILDER	48
4.1.1.	Diagrama del funcionament de la classe Builder	48
4.1.2.	Funció __init__	52
4.1.3.	Funció getAndCheckFileValues	53
4.1.4.	Funció checkRepeatedResidues	54
4.1.5.	Funció __addNewMolecule	55
4.1.6.	Funció __processResidue	56
4.1.7.	Funció __addNewResidue	57
4.1.8.	Funció convertxyz + convertxyz2	58
4.1.9.	Funció convertToAtoms	58
4.1.10.	Funció printErrors	59
4.1.11.	Funció getLast3M	59
4.1.12.	Funció PeriodicBoundaryCondition	59
4.1.13.	Funció __checkCollisionInSystem	60
4.1.14.	Funció __checkCollisionInSystem2	61
4.1.15.	Funció __getRandomPositionWithoutCollision	61
4.1.16.	Funció getRandomThetaAndPhi	62
4.1.17.	Funció getSystem	62
4.2.	INT2CAR_M	62
4.2.1.	Funció int2car_x	62
4.2.2.	Funció int2car_y	63

4.2.3.	Funció int2car_z.....	63
4.2.4.	Funció new_car	63
4.2.5.	Funció print_xyz.....	64
4.2.6.	Funció int2car	64
4.3.	FILES.....	64
4.3.1.	Funció __init__	65
4.3.2.	Funció readFile	65
4.3.3.	Funció storeFile	66
4.3.4.	Funció printFile	66
4.4.	FILEPDB.....	66
4.4.1.	Funció readFile	67
4.4.2.	Funció storeFile	67
4.4.3.	Funció addLine.....	68
4.4.4.	Funció addEndLine	68
4.5.	FILEPREPI	69
4.5.1.	Funció __init__	69
4.5.2.	Funció readFile	69
4.5.3.	Funció trataFile	70
4.5.4.	Funció getCar	70
4.5.5.	Funció getPar	70
4.5.6.	Funció checkFile.....	70
4.5.7.	Funció storeFile	71
4.6.	LINEPREPI	71
4.6.1.	Funció __init__	71
4.7.	MESSAGES	71
4.7.1.	Funció getTextMessage.....	71
4.8.	SYSTEM.....	72
4.8.1.	Funció __init__	72
4.8.2.	Funció addMolecule	72
4.8.3.	Funció printSystemNumberElements	73
4.8.4.	Funció printSystemMolecules	73
4.9.	MOLECULE.....	73
4.9.1.	Funció __init__	73
4.9.2.	Funció addResidue	74
4.9.3.	Funció printMolecule	74
4.9.4.	Funció getNextChainID.....	74
4.10.	RESIDUE.....	75

4.10.1. Funció __init__	75
4.10.2. Funció printResidue	75
4.10.3. Funció addAtom	75
4.10.4. Funció getAtom	75
4.11. ATOM	76
4.11.1. Funció __init__	76
4.11.2. Funció printAtom	76
4.11.3. Funció deleteDUMMLines	77
5. EINES UTILITZADES	78
5.1. Python	78
5.1.1. Història del Python	78
5.1.2. Programació amb Python	79
5.1.3. Llibreries de Python	79
5.1.4. Sci.py	80
5.1.5. Num.py	80
5.1.6. Sys.py	80
5.1.7. Pdb.py	80
5.1.8. Os.py	81
5.2. Py-Charm Edu	81
5.3. Jmol	82
5.4. Git	82
5.5. Tortoise	83
6. APLICACIÓ	85
6.1. Mode d'ús	85
6.2. Exemples d'aplicació	85
6.2.1. Preparació de les dades d'entrada	86
6.2.1.1. Fitxers de càrrega	86
6.2.1.2. Monòmers en format PREPI	88
6.2.2. Anàlisi dels resultats	96
6.2.2.1. Fitxers en format PDB	96
6.2.2.2. Visualització sistema	100
6.2.3. Simulació de la densitat	104
7. ANÀLISI DE L'IMPACTE AMBIENTAL	109
CONCLUSIONS	111

PRESSUPOST I/O ANÀLISI ECONÒMICA	113
BIBLIOGRAFIA	114
ANNEX A1 – MANUAL D'USUARI	117
A1.1 Introducció.....	117
A.1.2 Requeriments previs	119
A1.2.1 Instal·lació entorn desenvolupament Python.....	119
A1.2.2 Instal·lació del programa SuSi	120
A1.2.3 Directoris i fitxers de treball	122
A1.3 Processament de les dades	124
A1.3.1 Dades d'entrada	126
A1.3.2 Exemple d'execució del programa SuSi	133
A1.3.2.2 Pas 2: Obtenir ajuda	133
A1.3.2.3 Pas 3a: Generació del sistema en mode real	134
A1.3.3 Resultats i exemples d'aplicació.....	136
A1.3.4 Llista de possibles errors	142
ANNEX A2 – DETALL TÈCNIC: ATRIBUTS I FUNCIONS	145
A2.1 Classes principals.....	145
A2.1.1 SYSTEM	145
A2.1.2 MOLECULE	146
A2.1.3 RESIDUE	147
A2.1.4 ATOM.....	147
A2.2 Classes d'entrada i sortida.....	148
A2.2.1 FILES	150
A2.2.2 FILEPDB	150
A2.2.3 FILEPREPI.....	151
A2.2.4 MESSAGES	152
A2.3 Classes de processament	153
A2.3.1 __INIT__	153
A2.3.2 INT2CAR_M	153
A2.3.3 BUILDER	154
ANNEX A3 – VERSIÓ DE DISTRIBUCIÓ	156
A3.1 Setup.py	156
A3.2 Afegir paràmetres sdist al setup	156
A3.3 Executar setup.py	157

A3.4 Com instal·lar aquestes classes.....	157
ANNEX A4 – FITXERS UTILITZATS	158
A4.1 Fitxers de càrrega	158
A4.2 Fitxers d'entrada	158
A4.3 Fitxers sortida.....	158
A4.3.1 PDB coordenades periòdiques.....	158
A4.3.2 PDB sense coordenades periòdiques	158
A4.3.3 Taula evolució densitat	158

1. Introducció

1.1. Objectius del treball

Tenint en compte que la majoria dels polímers presenten una estructura amorfa, moltes vegades en forma de malla tridimensional, això fa que alhora de fer simulacions atomístiques d'aquest sistema sigui molt difícil la generació de la seva estructura inicial.

L'objectiu principal d'aquest projecte és l'anàlisi e implementació d'un programa informàtic que permeti generar estructures tridimensionals de polímers d'una manera generalitzada a partir de la seva composició química.

El programa hauria tenir les següents característiques:

- Hauria de servir com a base per la construcció tridimensional de materials per simulació tenint en compte la seva estructura 3D a escala molecular.
- Hauria de permetre simplificar la preparació de simulacions de polímers, y permetre la creació simple i de manera generalitzada d'un sistema de cadenes dels mateixos a partir de l'estructura coneguda del seus residus.
- Empaquetar tots les cadenes polimèriques dins d'una caixa rectangular ortoròmbica de tres dimensions de manera amorfa i amb una densitat el més propera possible al valor real
- No distorsionar gaire l'estructura inicial del monòmer per tal d'encabir-lo dins de la caixa periòdica de simulació.
- Retornar com a sortida un fitxer en format PDB amb la composició del sistema amorf creat.
- Testejar la viabilitat del programa a partir de varies generacions de sistemes polimèrics amorfs de complexitat creixent. Una vegada obtingut els sistemes s'hauria de procedir a la seva relaxació per simulació atomística a pressió constant fins a obtenir la densitat final més estable del sistema obtingut i poder-lo comparar amb el seu valor experimental.

1.2. Abast del treball

En primer lloc, en aquest capítol es farà una breu descripció dels polímers, el seu origen, de quins tipus hi ha i quina és la seva estructura.

Els polímers són molècules extremadament grans que són essencials per a la nostra existència; i es tracta d'un compost químic, natural o sintètic, format per polimerització i consisteix essencialment en unitats estructurals repetides. Els polímers són materials d'alt pes molecular, això és degut al fet que

estan formats per la unió de molècules més petites anomenades monòmers, que s'uneixen entre si formant cadenes que poden presentar ramificacions o entrecreuaments mitjançant forces de Van der Waals, ponts d'hidrogen o interaccions hidrofòbiques. Poden arribar a ser molt grans, pel que se'ls anomena també macromolècules. Normalment són orgàniques, unides entre si per enllaços covalents i formades en reaccions de polimerització. La unitat estructural que es repeteix al llarg de tota la cadena polimèrica es coneix com a unitat repetitiva.

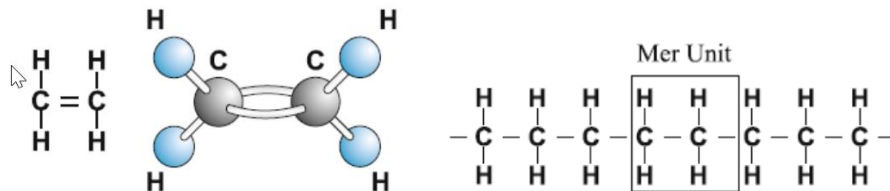


Figura 1.2.1. Exemple de monòmer etilè i de polímer^[1].

1.2.1. Origen dels polímers

El terme “polímer” (que prové del grec “polys” que significa “molt” i “meros” que significa “part”) va ser introduït el 1833 pel químic suec “Jöns Jakob Berzelius. Tot i que només havia passat un any des que va introduir el terme “isòmer” (també del grec “isos” que significa “igual”) per descriure substàncies que tenen composicions idèntiques però propietats diferents, va veure que era necessari un nou terme per distingir entre dos possibles tipus d'isomerisme entre dos tipus de substàncies:

- Quan tenien fórmules de composició absoluta idèntiques en què la diferència de propietats era atribuïble a una diferència en l'ordenació dels grups component atòmics (format d'etil i acetat de metil).
- Quan tenien fórmules de composició relativa idèntica, però diferents fórmules de composició absoluta en què la diferència de propietats era atribuïble a polimerització o diferència en el nombre total d'àtoms presents (etè i butè).

Com el significat original del terme polímer no estava relacionat en cap forma amb la mida, el 1922 el químic alemany Hermann Staudinger, va considerar necessari introduir la paraula “macromolècula” per descriure grans molècules de cadenes orgàniques unides de manera covalent que contenien més de 103 àtoms.

1.2.2. Classificació dels polímers

Hi ha diverses maneres de classificar els polímers d'acord amb la seva estructura molecular; però sempre es poden distingir entre dos tipus generals com *termoplàstics* i *termoestables*.

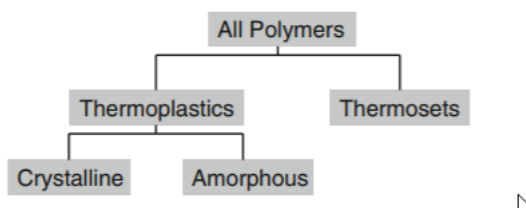


Figura 1.2.2.1 Classificació simple dels polímers^[1].

La diferència fonamental entre els dos tipus té a veure amb la seva estructura química i amb l'enllaç entre les cadenes moleculars; el que afecta les propietats dels materials i per tant a les seves característiques tèrmiques i de processament generals.

1.2.2.1. Termoplàstics

Els polímers *termoplàstics* o *lineals* sols tenen enllaços secundaris entre cadenes i tenen la característica de poder-se fondre o modelar; ja que en funció de la temperatura en què es trobin d'acord amb les temperatures de referència per la transició de solidificació (T_g) i la temperatura de fusió (T_m) poden existir en qualsevol de les tres fases: sòlid, flexible o viscos. En general, aquests polímers són més fàcils de produir i costen menys que els termoestables.

Els termoplàstics es poden dividir en amorfs o cristal·lins. Normalment el grau de cristal·lització és baix en metalls cristal·lins, ceràmiques i altres materials; i per tant els polímers rarament són més del 50% cristal·lins. La característica dels polímers cristal·lins és que solen ser més densos que els amorfs a causa de l'empaquetatge més proper de les seves molècules de cadena llarga i, en general, es milloren les següents propietats: fricció i desgast, duresa, menys fluïdesa/dependència del comportament en funció del temps, resistència a la corrosió i / o resistència a l'estrès ambiental i esquerdes.

1.2.2.2. Termoestables

Els termoestables o reticulats tenen enllaços primaris i secundaris entre cadenes; i no es poden fondre ni modelar en el sentit general del terme, ja que solen sols existir com a sòlids o flexibles.

Existeixen dos tipus de termoestables: Els polímers molt reticulats que gaudeixen d'una alta estabilitat tèrmica i dimensional; i aquells lleugerament reticulats que sovint es denominen *elastòmers*, ja que tenen una gran flexibilitat i es deformen, però quan cedeix la força tornen al seu estat original de polímer i evita que aquestes quedin enganxades entre elles.

1.2.3. Estructura dels polímers

Estructura dels polímers

Per tenir en compte l'estructura dels polímers, cal considerar el nivell químic i el físic. Com a estructura química es tindrà en compte la construcció de la molècula individual i com a estructura física a l'ordenament d'unes molècules respecte a altres.

1.2.3.1. Estructura química dels polímers

Com s'ha comentat, en els polímers la unió entre monòmers es realitza a partir d'enllaços covalents. Les forces responsables de cohesió entre les diferents cadenes de monòmers que formen el polímer poden ser de naturalesa molt diversa i es troben fortament condicionades a les característiques dels àtoms de la cadena principal. La polaritat i el volum dels àtoms afectarà especialment les forces de cohesió entre cadenes que al mateix temps també determina la flexibilitat del material, les temperatures de referència i la capacitat de cristal·litzar. En general com major sigui la polaritat de les molècules i per tant la força de cohesió entre les cadenes, es pot dir que més rígida resultarà el polímer; i per tant més alta serà la temperatura de fusió.

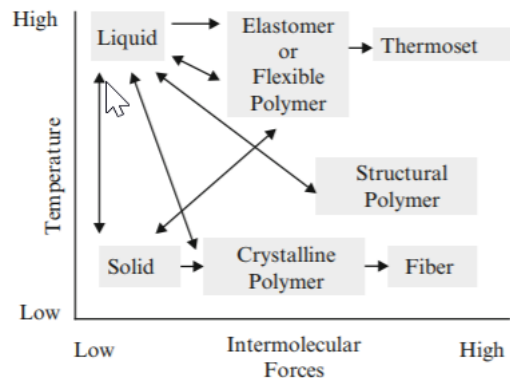


Figura 1.2.3.1.1. Gràfica de temperatura versus forces^[1].

En l'estructura dels polímers cal tenir en compte el tipus d'unió entre els diferents monòmers, i per tant la polimerització que ha sofert la macromolècula. S'anomena Polimerització la reacció química en què dues molècules es combinen per formar-ne un altre en la que es repeteixen unitats estructurals de les primeres amb igual composició percentual quan aquestes són iguals.

Hi ha diferents tipus de polimerització, segons el mecanisme pel qual s'uneixen estructures monomèriques o segons les condicions experimentals de reacció. Els podem classificar de la següent manera:

- *Per addició*: Té lloc mitjançant la interacció d'enllaços múltiples presents en les molècules del monòmer. El polímer resultat té la mateixa composició que el monòmer del qual procedeix i no existeixen diferències entre les posicions relatives dels àtoms en les molècules insaturades de monòmer i en les unitats estructurals del polímer. Pot ser per suspensió, per emulsió i per addició en massa.
- *Per condensació*: Els monòmers sempre s'uneixen els uns amb els altres de la mateixa forma i difereix de l'anterior en el fet que té lloc la pèrdua d'una molècula senzilla perquè es dugui a terme i no exigeix la presència d'enllaços insaturats de monòmers. Els grups reactius que acostumen a propiciar la reacció són hidroxil (-OH), carboxil (-COOH), carbonil (-CO). Per aquest mètode es poden obtenir tant macromolècules de pes elevat i finit (lineals) com estructures de xarxes tridimensionals amb pes molecular (reticulars)
- *Altres*: Per creixement d'etapes on la cadena s'incrementa en més d'un monòmer; o per creixement de cadena on els monòmers passen a formar part de la cadena d'un en un.

El pes molecular influeix en moltes propietats físiques dels polímers com ara resistència mecànica, elasticitat, temperatures de referència i per tant sobre el seu estat d'agregació. Tenint en compte que la majoria dels polímers estan formats per una barreja de molècules que tenen diferent grau de polimerització; i per tant diferent pes molecular caldrà treballar amb valors mitjans com ara la mitja aritmètica " M_n " o quadràtica " M_w ", on " N_i " són el nombre de moltes de les espècies " i " pes molecular " M_i "

$$M_n = \frac{\sum N_i * M_i}{\sum N_i}$$

$$M_w = \frac{\sum N_i * M_i^2}{\sum N_i * M_i}$$

L'heterogeneïtat d'un polímer es pot mesurar per la relació entre les dues mitjanes " M_w/M_n " i normalment es coneix com l'índex de polidispersitat i sol trobar-se sobre 1,5/3.

Quan en les reaccions de polimerització no hi ha reaccions secundàries normalment s'obtenen polímers lineals amb o sense ramificacions que poden ser curtes o llargues; i les propietats dels polímers varien considerablement en funció d'aquestes i de la seva longitud.

A partir d'aquestes estructures, quan certes cadenes s'uneixen amb altres veïnes a partir d'altres cadenes d'igual o diferent naturalesa i s'obté una xarxa tridimensional, és quan s'obtenen polímers reticulats; i és necessari que cada molècula s'uneixi a dos o més punts de la resta de molècules.

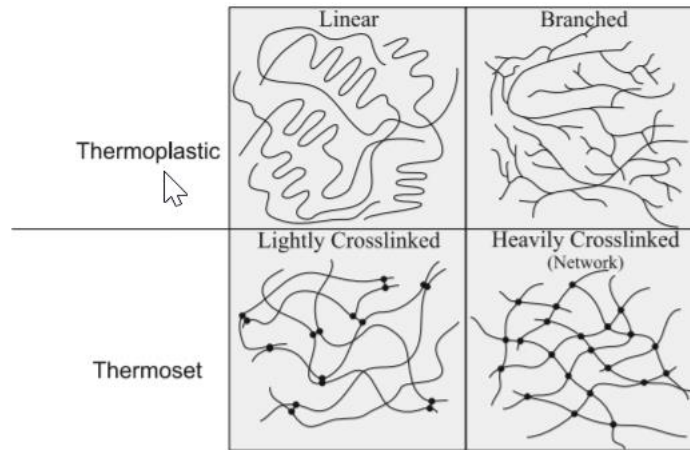


Figura 1.2.3.1.2. Exemple de polímers lineal, ramificat i reticulat^[1].

1.2.3.2. Estructura física dels polímers i cristal·lització

Les regions on les cadenes de polímer existeixen en una matriu ordenada es diuen dominis cristal·lins; en canvi les regions on les cadenes existeixen d'una manera menys ordenada es denominen com regions amorfes. Per tant, un polímer en funció de la seva capacitat de cristal·litzar podrà ser amorf o cristal·lí.

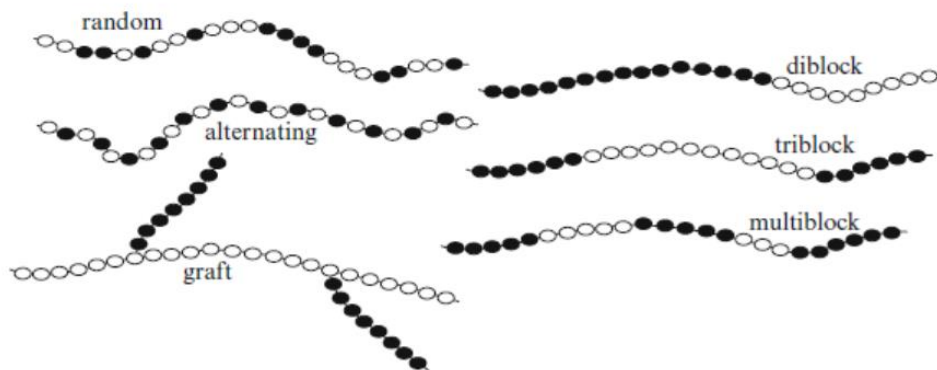


Figura 1.2.3.2.1. Exemple de polímers lineal, ramificat^[1].

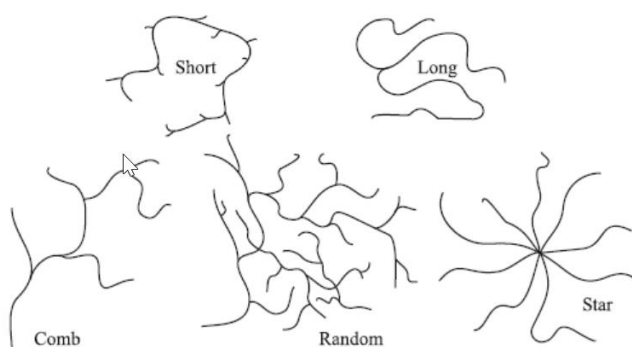


Figura 1.2.3.2.2. Exemple de polímers reticulat^[1].

La quantitat i tipus de cristal·lització depèn tant de l'estructura del polímer com del seu tractament. Per exemple, la proporció de cristal·lització es pot controlar controlant la velocitat de formació d'àrees cristal·lines; i la refrigeració a una velocitat més lenta permet que les cadenes es dobleguin, etc. donant un producte amb un major grau de cristal·lització.

Existeixen diversos factors que sempre poden influir en la forma que pot adoptar un polímer:

- La flexibilitat de la cadena està relacionada amb les energies d'activació necessàries per iniciar moviments segmentaris rotacionals i vibracionals. Per a alguns polímers, a mesura que augmenta la flexibilitat, la tendència a la formació cristal·lina augmenta. Els polímers que contenen enllaços C-C, C-N i C-O separats periòdicament permeten canvis ràpids que contribueixen a la flexibilitat de la cadena polimèrica i a la tendència cap a la formació cristal·lina. Tanmateix, la rigidesa de la cadena també pot millorar la formació cristal·lina en permetre o fomentar que només es produeixin certes conformacions "ben ordenades" dins de la cadena de polímers.
- La cristal·lització es veu afavorida per la presència d'unitats separades periòdicament que permeten associacions fortes entre les cadenes.
- La regularitat estructural i l'absència de grans substituents augmenten la tendència a la cristal·lització. L'efecte precís dels substituents depèn d'una sèrie de factors que inclouen la ubicació, la mida, la forma i les interaccions recíproques. Per exemple, la presència d'etil a substituents d'hexil tendeix a disminuir la tendència a la cristal·lització, ja que la seva contribució principal és augmentar la distància mitjana entre cadenes i, per tant, disminuir les forces de connexió secundàries. Quan els substituents es tornen més llargs (de 12 a 18 carbonis) i es mantenen lineals, es pot produir un nou fenomen: la tendència de les cadenes laterals a formar dominis cristal·lins.

Les temperatures de referència per la transició vítria i la de fusió són fonamentals per poder avaluar el volum dels polímers. Per això, en la figura de la gràfica 1.3.2.2 es representa com a exemple la variació del volum específic d'un polímer respecte a la temperatura.

Els termoplàstics a temperatures elevades normalment es troben en estats fos o líquid en molt de moviment i amb un volum lliure (no ocupat) associat a les molècules gran, que comporta un volum específic elevat. A mesura que la temperatura disminueix i hi ha menys energia disponible, els canvis també disminueixen gradualment amb el volum específic; fins que d'acord amb les lleis de la termodinàmica a certes temperatures es procedeix a ordenar les molècules formant estructures cristal·lines sòlides.

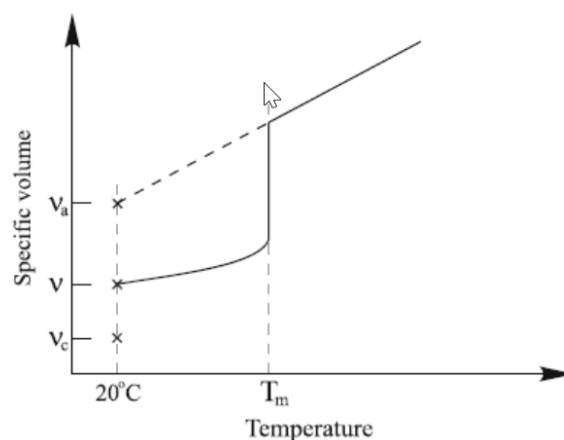


Figura 1.2.3.2.3. Gràfica de volum específic versus temperatura ^[1]

En les gràfiques de volum versus temperatura cal tenir en compte la temperatura de cristal·lització " T_c " que seria la necessària per a realitzar un canvi d'estat cristal·lí a amorf; i la temperatura de fusió " T_m " que es dona quan es canvia de sòlid a líquid.

En els polímers amb estructures complexes i irregulars que presenten viscositats molt elevades en estat líquid, encara que es disminueixi la temperatura, no es pot aconseguir que es cristal·litzi i segueixen sempre amb les conformacions desordenades més pròpies dels líquids. Per aquests casos la disminució del volum es produeix de forma gradual juntament amb la temperatura; i cal tenir en compte una nova temperatura de referència, que seria la de transició vítria " T_g " amb la que es sofreixen grans canvis en les propietats, com ara limitació de moviments.

Tenint en compte que un polímer pot tenir regions amorfes i cristal·lines, quan aquests es calenta sempre s'aconsegueix augmentar primer la mobilitat de les molècules de les zones amorfes; ja que les zones cristal·lines seguiran sòlides fins a arribar al punt de fusió dels cristalls on s'aconseguirà una mobilitat total de tota l'estructura.

1.2.4. Matrius polimèriques i simulacions

Tot i que els polímers presents i fonamentals tant en la nostra vida diària com en diferents àrees de la indústria, tal com hem vist per la seva estructura, en algunes ocasions encara és complex comprendre i preveure el seu comportament a escala molecular.

Per poder realitzar prediccions sobre el seu comportament al llarg de la història s'han desenvolupat diversos mètodes. Per exemple es van utilitzar les tècniques de simulacions de dinàmica molecular o "MD" per estudiar els polímers, ja que aquestes proporcionen informació important sobre l'estructura molecular i el moviment a escala atòmica. Podríem posar com a exemples diferents estudis de Carlos Alemán que ha documentat en varis articles publicats en la revista "JOURNAL OF CHEMICAL PHYSICS" sobre simulacions de dinàmica molecular com ara *"Molecular Dynamics Simulation Study of Methanesulfonic Acid o Modeling the Tetraphenylalanine-PEG Hybrid Amphiphile: From DFT Calculations on the Peptide to Molecular Dynamics Simulations on the Conjugate"*^[2-4]

A mesura que van transcorrent els anys, i la potència de la informàtica augmenta de forma exponencial, és possible anar desenvolupament noves metodologies de simulació cada cop més efectives i afegint noves complexitats.

Es poden enumerar diferents estudis com ara el de les interaccions de polímers amb nano tubs de carboni d'una sola paret que van realitzar els xinesos *Wei Liu, Chuan-Lu Yang, Ying-Tao Zhu, and Meishan Wang*^[5] la predicció de la solubilitat dels compostos farmacèutics en els portadors polimèrics en la universitat de Philadelphia o l'estudi de l'estructura i la mecànica de polímers vidriosos realitzat per *Jean-Louis Barrat, Jörg Baschnagel i Alexey Lyulin*^[6]. Malgrat tots aquests avenços exitosos, els polímers segueixen sent difícils de modelar i d'estudiar.

Tenint en compte que polímers són macromolècules que poden estar compostes de cadenes molt llargues que poden ser d'un únic tipus de monòmer o diferents, a més de cadenes de copolímer trifàsic, cadenes ordenades aleatòriament, estrelles, anells, barreges compostes de diferents longituds de cadena, diferents monòmers de ramificació...; la simulació d'un sistema polimèric de forma realista és summament complex amb un elevat cost computacional, si no quasi impossible.

Per tant, perquè el polímer sigui tractable i es pugui estudiar molts cops s'ha de simplificar i realitzar les simulacions d'aquest sobre regions més reduïdes, cadenes mitjanes o composicions simplificades que siguin representatives de les propietats a estudiar.

Per a poder realitzar aquestes modelitzacions sobre polímers es poden utilitzar diversos programes comercials normalment gratuïts que permeten construir un polímer únic estès segons una seqüència predefinida.

Alguns programes implementen estratègies per generar cadenes més relaxades. Aquestes estratègies van des de permetre la manipulació de l'usuari dels angles dièdrics d'una cadena (p. Ex., Pymol, Polymer Modeler)^[7-8] fins a l'explotació de sistemes basats en Monte Carlo per relaxar la cadena (p. Ex., Monte Carlo millorada)^[9]. Altres programes no només són capaços de construir una sola cadena, sinó també de muntar un sistema polímer complet (p. Ex., Moltemplate, Monte Carlo millorat, Materials Studio)^[10] i de vegades permet assignar paràmetres des d'un camp de força disponible.

2. Estat de l'art

2.1. Antecedents

Tot i que les propietats físiques dels polímers estan molt relacionades amb la seva estructura química i amb l'ordenació espacial de les cadenes macromoleculars en l'embalatge; degut a les altes densitats dels polímers amorfs i la complexitat que presenta la modelització de la connectivitat de matrius polimèriques reticulades, és difícil disposar de les estructures tridimensionals inicials per poder realitzar prediccions mitjançant simulacions per ordinador.

Al llarg dels anys s'han anat dissenyant diverses tècniques computacionals molt ràpides que podien predir certes propietats físiques dels polímers tenint en compte sols la seva estructura; com ara les proposades per *Fedors, van Krevelen* i *Bicerano*^[11-13]. Però el fet de només tenir en compte les propietats físiques d'estructura dels polímers donen resultats poc realistes quan s'apliquen a sistemes en què la constitució química i/o les condicions experimentals són diferents de les utilitzades en la parametrització.

Altres mètodes computacionals més comuns utilitzen algorismes que permeten l'empaquetatge de cadenes a densitat experimental de forma aleatòria juntament amb minimització d'energia o creixement de cadenes de Monte Carlo basades en estats isomètrics rotacionals (RIS).^[14]

L'inconvenient que tenen tots és el fet que es van dissenyar per generar “un nombre reduït d'estructures polimèriques”; de forma que servissin com a punt de partida per a les simulacions Monte Carlo (MC) o de dinàmica molecular (MD). I els algorismes MC i MD, per si mateix, no són capaços de generar estructures prou denses i amorfes.

2.2. Nou mètode de simulació d'estructura (SuSi)

L'any 2003 el David Curco i el Carles Alemany^[15], des del departament d'Enginyeria Química de la Universitat de Barcelona i de la Universitat Politècnica de Catalunya respectivament, van publicar un article on es presentava una nova estratègia computacional per generar estructures atòmiques representatives de polímers densos.

El nou mètode alternatiu anomenat SuSi (simulació d'estructura) s'organitza en un procediment de dos passos independents i ben definits. En primer lloc, s'aconsegueix la generació de les estructures mitjançant un algorisme que minimitza la tensió de torsió d'enllaços; i a continuació, en un segon pas mitjançant l'aplicació d'un algorisme iteratiu, s'aconsegueix la relaxació de les posicions atòmiques

d'aquestes estructures tot eliminant les col·lisions entre partícules no enllaçades (diferents parts de la cadena polimèrica o entre diferents cadenes). A l'estar compost aquest mètode de dos procediments independents requereix menys poder computacional que altres estratègies prèviament desenvolupades; i és capaç de produir gran quantitat de polímers completament diferents sense superposicions atòmiques ni de tensions torsionals.

El mètode es pot utilitzar per obtenir configuracions inicials per a simulacions de Monte Carlo (MC) i de Dinàmica Molecular (MD), però també permet generar una sèrie d'estructures independents per a l'avaluació directa de propietats físiques.

En l'article, on es presentaven els resultats obtinguts, es van considerar tres polímers completament diferents; i es va poder demostrar que "SuSi" proporcionava estructures sense tensió torsional ni superposicions atòmiques.

En aplicar aquest mètode per calcular la solubilitat de petites molècules de gas en polietilè amorf "PE" (polímer de baixa densitat = $0,85 \text{ g/cm}^3$) es va comprovar que els resultats proporcionats pel nou mètode "SuSi" coincidien totalment amb les dades experimentals. Al mateix temps també va donar bon resultat la comparació de les estructures generades examinant les conformacions resultants amb altres obtingudes per altres mètodes.

En calcular la densitat de l'energia de cohesió i els paràmetres de solubilitat d'Hildebrand pels sistemes d'àcid poli (glicòlic), PGA i àcid polilàctic, L,D-PLA (polímers d'alta densitat= $1,50$ i $1,25 \text{ g/cm}^3$, respectivament), també es van obtenir resultats molt satisfactoris; i es van generar empaquetatges representatius exitosos amb recursos computacions mínims.

2.3. Estudi sobre el rendiment

Un mica més tard, es va realitzar un nou estudi per investigar el rendiment del nou mètode SuSi^[16], que es va centrar en la revisió de la influència dels paràmetres que es troben relacionats amb la mida del sistema i en la mateixa generació d'aquest.

El mètode SuSi es veu influenciat per alguns paràmetres que s'han d'escollir prèviament a la realització de cada un dels càlculs; i l'objectiu és avaluar el seu impacte i poder proporcionar certes advertències per poder utilitzar SuSi de forma més eficaç.

SuSi considera que un sistema està format per "N" cadenes de polímer de "n" partícules, que es van col·locant en una caixa cúbica amb condicions periòdiques de contorn, on el volum de la caixa es defineix en funció de la densitat del sistema polimèric a generar.

Les microestructures es generen mitjançant un algoritme per obtenir una tensió en la torsió d'enllaç mínima. Això s'aconsegueix per mitjà d'una reducció dels radis de les partícules $\{R\}$, que es multipliquen per un factor $\lambda < 1$. Els radis escalats $\{\lambda * R\}$ s'utilitzen per a créixer les cadenes de manera seqüencial, partícula per partícula. Les parts principals del procediment utilitzat per generar les coordenades de les partícules $n * N$ del sistema es poden resumir de la manera següent:

- Tres partícules enllaçades de la primera cadena es col·loquen en posicions arbitràries dins del quadre de simulació.
- Per a la quarta partícula generen de forma aleatòria un nombre “k” de posicions.
- Es calcula l'energia associada a cada posició $\{E_j\}$ on “j” varia d'1 a “k”; i es tria a través d'un criteri d'acceptació Monte Carlo una posició.
- Es generen “k” posicions per a la cinquena partícula. Aquestes posicions es classifiquen com inviabilitats o factibilitats depenent si hi ha superposicions atòmiques o no entre els àtoms separats per més de tres enllaços. Es valora l'energia de les posicions factibles “j” on $(j \leq k)$, mentre que les posicions no factibles es descarten.
- El punt anterior es repeteix per a les partícules restants de la cadena. Si per a una partícula i per totes les seves posicions generades es detecten superposicions atòmiques, aleshores la cadena es reconstrueix, tenint en compte un augment amb la quantitat d'errors.
- Les coordenades de les “n” partícules de les pròximes cadenes es generen utilitzant el mateix procediment amb la diferència que els tres primers àtoms es troben a partir d'un procediment de cerca aleatòria.

El segon pas consisteix en la millora de les estructures generades a través d'un algoritme de relaxació. El mètode, que requereix un mínim de recursos computacionals, es pot resumir de la manera següent:

- L'energia de l'estructura generada en el primer pas, $E_{initial}$, s'avalua tenint en compte les electrostàtiques, van der Waals i les interaccions torsionals entre les partícules $n * N$. Si $E_{initial}$ és superior al valor llindar especificat per l'usuari, l'estructura es descarta directament i no s'aplica cap algoritme de relaxació. D'aquesta manera, l'algoritme de relaxació no s'aplica a les microestructures menys afavorides, la qual cosa permet estalviar considerable esforç a l'ordinador.
- L'energia del sistema es minimitza. Les expressions analítiques s'utilitzen per determinar les noves posicions de les partícules i corregir aquestes posicions per tal de satisfer les restriccions de la geometria interna. L'eficiència d'aquest procés es millora mitjançant l'ús d'una estratègia de tall doble per a les contribucions electrostàtiques i van der Waals, és a dir, el tall s'incrementa després d'alguns passos de minimització.

Per tant, es pot dir que la combinació dels algorismes de generació i relaxació descrits anteriorment permet definir SuSi com un generador aleatori d'energia mínims, i l'estratègia utilitza un mètode aleatori per construir microestructures amb una tensió mínima d'enllaç i sense superposició atòmica.

Com a base d'aquest segon estudi, els càlculs es realitzen sobre dos polímers comercials importants: en àcid L,D- polí làctic i el niló; i, es van voler examinar els efectes que podien produir els paràmetres que s'utilitzen per definir les característiques de l'algoritme de generació del mètode SuSi:

- “N” i “n” que es troben relacionats amb les dimensions del sistema a modelar; ja que són el Nombre de polímers i el nombre de partícules
- “λ” i “k” amb l'algoritme; ja que són el factor d'escala utilitzat, i el nombre de posicions que es generen de forma aleatòria.

Com a conclusions es va poder observar que tot i que el mètode SuSi va néixer per a treballar amb polímers d'alta densitat, els resultats són pràcticament independents d'aquesta propietat; però en canvi altres factors sí que varien:

- El temps augmenta en augmentar la mida del sistema i del factor d'escala “λ” que es vol crear; en canvi disminueix quan “k” arriba a un valor llindar (14).
- Els models són energèticament representatius quan les següents variables es troben en els rangs “λ=0,725-0,75” i “k=8-14”. Aquests valors requereixen una quantitat considerable de recursos computacionals i/o proporcionen microestructures amb energies d'interacció repulsiva.
- Per a una determinada quantitat d'unitats, els resultats proporcionats per SuSi són gairebé independents de “N” i “n”. Cal tenir en compte que quan “N” és molt gran i “n” molt petit es produeix una excepció.

2.4. Estratègia teòrica per proporcionar models atòmics de polímers comb-like

El juny del 2004 es va presentar una nova estratègia computacional anomenada SuSi/CB (CB-configurational bias)^[17] per modelar la fase amorfa dels polímers comb-like; ja que es va considerar que l'estudi de polímers sintètics capaços d'adoptar estructures supramoleculares mereixia una atenció especial.

La síntesi, estructura, propietats i aplicacions de polímers comb-like sempre han estat àmpliament debatuts; però l'interès principal d'aquests sistemes macromoleculars resideix en el seu comportament d'organització pròpia.

Els polímers compostos consisteixen en llargues cadenes laterals flexibles, típicament grups lineals, units a una cadena principal rígida; i les evidències experimentals prèvies a aquest estudi recollides per a diferents famílies de polímers comb-like sempre indicaven que l'estructura d'aquests sistemes en estat sòlid es regeix principalment per la forta segregació del domini de la cadena principal sobre la cadena lateral, que seria el mateix que a la separació de micro fases per copolímers en bloc.

L'estructura bifàsica adoptada per polímers comb-like quan la cadena principal i les cadenes laterals es cristal·litzen ha estat ben caracteritzada per la calorimetria d'escaneig diferencial, la polarització infraroja, la radiografia i la difracció d'electrons. En aquest punt, les cadenes laterals helicoidals rígides es recullen en capes i les cadenes laterals flexibles que emanen de les cadenes principals cristal·litzen en una fase separada omplint l'espai entre les capes.

En el rang de temperatura adequat, tots aquests polímers formen mesofases, en què l'estructura altament ordenada segueix sent indestructible, però les cadenes laterals es fonen. A més, en alguns casos aquesta última mesòfase es transforma escalfant-se en una estructura en la qual les cadenes principals rígides abandonen l'estructura en capes i les cadenes laterals estan en un estat completament desordenat.

Com les estructures supramoleculares de polímers comb-like són sistemes densos que es troben ordenats heterogèniament, era extremadament difícil realitzar una simulació atòmica de per les tècniques convencionals de Monte Carlo (MC) i dinàmica molecular. Així i tot, en altres recerques ja es va presentar una estratègia molt eficaç per estudiar l'estructura bifàsica de polímers comb-like utilitzant simulacions MC fora de la xarxa^[18-21]. L'estratègia es basava en la combinació de l'algoritme Monte Carlo (CB) i un mètode basat en càlculs de mecànica molecular per estudiar sistemes ordenats.

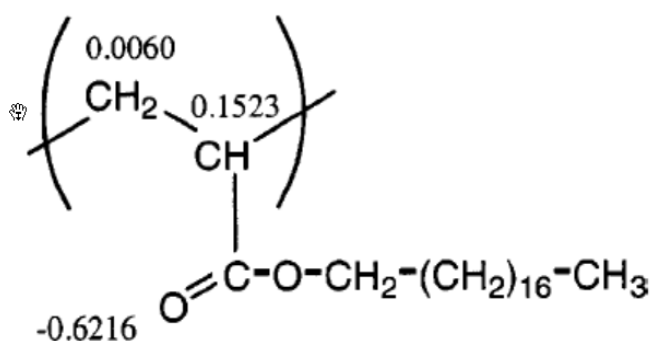
Com qualsevol polímer convencional, els polímers comb-like són materials semi cristal·lins amb més d'un 50% de fase amorfa; però en aquest cas cal tenir en compte que la fase amorfa es caracteritza per la falta d'ordre en la conformació tant de les cadenes principals com de les laterals. Per tant, com la fase amorfa afecta directament sobre les propietats dels sistemes, hi havia molt poca informació experimental disponible sobre l'organització de les molècules; i la segregació observada a les estructures supramoleculares de polímers comb-like, que està molt relacionada amb les propietats d'aquests materials, podria estar absent en la fase amorfa pel que es refereix a les unitats que no estan contingudes en les cadenes.

L'objectiu d'aquesta nova estratègia del 2004 SuSi/CB^[17] era presentar una estratègia computacional per modelar la fase amorfa dels polímers comb-like. En aquesta metodologia es va voler combinar la simulació d'estructura SuSi (procediment desenvolupat recentment per generar models atòmics fiables de polímers durs amorfs) i un nou algoritme CB-Monte Carlo. L'estratègia resultant permet generar estructures representatives de polímers comb-like, és a dir, estructures sense superposicions atòmiques i tensió mínima torsional; i poder analitzar en detall la influència de diferents paràmetres per determinar tant la fiabilitat de les estructures que es generen com per mesurar l'esforç computacional requerit.

Es aquell treball es va demostrar que el mètode SuSi/CB era perfecte per la simulació de polímers complexos atomisticament detallats en la fase amorfa. Aquest procediment consisteix en un algoritme que permet la generació d'estructures atòmiques amb una mínima tensió torsional, que posteriorment es relaxen utilitzant el mètode avançat de Monte Carlo CB. Aquest mètode (MC) és el mateix que es va demostrar molt eficient per tractar cadenes laterals llargues de polímers complexos.

Si es comparen i busquen les diferències més importants respecte als mètodes detallats anteriorment de SuSi serien que el nou mètode SuSi/CB utilitza dos coeficients d'escalat (λ_m i λ_s) per generar les estructures mentre que l'anterior SuSi sols n'utilitzava un; i que en aquest segon mètode es requereix un sofisticat algoritme de relaxació per a polímers comb-like, mentre que anteriorment la minimització que es va utilitzar per a polímers lineals era molt simple. Per últim el nou mètode va requerir la implementació d'un coeficient d'escalat (λ_{CB}) en l'algoritme Monte Carlo CB per relaxar de manera més eficient els polímers complexos en la fase amorfa, que no va ser necessari quan el sistema simulat estava organitzat en estructures bifàsiques.

L'estudi per avaluar la fiabilitat de SuSi/CB, i la influència dels factors λ_m , λ_s i λ_{CB} , o de la densitat del sistema, es va fer amb simulacions de poly (octadecil acrilat) o PA-18. També es va tenir en compte la influència de les interaccions electrostàtiques; i es va poder comprovar que no tenen afectació en la relaxació de les estructures resultants. Els resultats van indicar que, en general, $\lambda_m = 0,715$, $\lambda_s = 0,675$ i $\lambda_{CB} = 0,75$ proporcionen els millors resultats i també són els més eficients des del punt de vista computacional. D'altra banda, els resultats obtinguts pel PA-18 amorf amb densitat $\rho = 1.085$ i 0.85 g / cm^3 revelen que la cadena principal presenta una rigidesa considerable, mentre que les cadenes laterals formen un grup parafínic no estructurat. Els resultats obtinguts tenint en compte les interaccions electrostàtiques eren gairebé idèntiques a les proporcionades quan aquestes interaccions es descuidaven, i per tant es va demostrar que el terme d'energia de Coulomb no era necessari per simular polímers complexos amb una distribució de càrrega similar a la del PA-18.



Imatge 2.4.1 Esquema PA – 18^[17]

2.5. influència del mètode de relaxació

Per finalitzar, l'octubre del 2004^[22] es van presentar dos mètodes alternatius per relaxar els models atòmics de polímers amorfs generats per la simulació d'estructura i es van comparar amb l'algoritme de minimització implementat originalment en SuSi. Aquests mètodes de relaxació estan basats en els aspectes geomètrics de les tècniques convencionals (CB) i de rotació concertada Monte Carlo (ConRot).

L'algorisme de generació implementat a SuSi es va demostrar molt potent i eficient, podent produir models atòmics amb una tensió mínima torsional utilitzant recursos informàtics insignificants; però també es van detectar algunes limitacions greus com ara que la memòria de la microestructura produïda per l'algorisme de generació no es perdia durant el procés; i per tant les característiques conformacionals generals de les cadenes individuals respecte a la microestructura inicial canviaven molt poc. I a més, l'efectivitat del mètode es va demostrar relativament baixa; ja que la millora de l'energia solia ser inferior al 10%.

Per això en aquest darrer treball, l'algoritme de minimització utilitzat en SuSi es va reemplaçar per estratègies geomètriques basades en tècniques de biaix configuracional (CB) i de rotació concertada (ConRot). Per examinar i comparar l'eficàcia i l'eficiència dels nous algorismes de relaxació, es van generar una sèrie de microestructures de polietilè i es va analitzar la qualitat dels seus models després de la relaxació.

Per realitzar les simulacions es van considerar dos sistemes polimèrics: quatre cadenes de polietilè on cadascuna contienien 400 grups CH₂ i vuit cadenes de polietilè amb 800 grups CH₂. El procediment basat en ConRot va demostrar-se el més eficient dels tres algorismes de relaxació que s'estaven analitzant, mentre que la minimització va resultar ser la menys eficient. D'altra banda, cal tenir en compte que el mètode basat en CB està limitat per les dificultats en la relaxació dels segments interiors de les molècules de cadena.

Com a conclusió després de descriure i analitzar diverses estratègies per relaxar els models atòmics de polímers amorfs generats per SuSi per realitzar els càlculs en dos sistemes polimèrics formats per cadenes de polietilè; es va veure que SuSi/ConRot i SuSi/Min eren els mètodes més eficaços, mentre que SuSi/CB estava limitat per les dificultats en la relaxació dels segments interiors de les cadenes. Cal assenyalar que l'eficiència dels mètodes SuSi/ConRot i SuSi/CB respecte a SuSi/Min probablement està relacionada amb la capacitat dels mètodes anteriors de mostrar l'espai configuracional. L'examen de les estructures relaxades per PE4-400 i PE8-800 va indicar que la influència de l'algorisme de relaxació en les propietats estructurals a gran escala de les cadenes polimèriques és gairebé insignificant. El motiu era que algorismes de relaxació actuaven sobre les interaccions desfavorables no vinculants, que resultaven d'unes petites superposicions i que es minimitzaven aplicant canvis molt petits a les coordenades atòmiques. Per tant, els procediments de relaxació no eren capaços d'induir canvis conformationals a les cadenes a causa de la naturalesa polimèrica dels sistemes. En canvi, l'efectivitat del mètode utilitzat per a la relaxació afectava considerablement les propietats relacionades amb l'energia.

Per últim es va concloure que l'energia després de la relaxació depenia de l'energia abans de la relaxació; i per tant, les estructures inicials de baixa energia donaven lloc sistemàticament a estructures relaxades de baixa energia. Els dos passos implicats en SuSi / ConRot donaven com a resultat una eficiència de l'algoritme de generació molt alta però també requerien un gran esforç computacional per dur a terme relaxació. Totes aquestes característiques van suggerir que una bona estratègia per modelar polímers amorfs consistia en la generació d'una gran quantitat d'estructures, la posterior categorització en termes d'energia i la relaxació de només aquells amb una energia menor que un cert tall.

3. Anàlisis

3.1. Necessitat del programa

Per la creació d'aquest projecte s'ha escollit el llenguatge Python ^[23] i la seva programació s'ha realitzat seguint l'esquema de programació orientada a objectes (Object Oriented Programing, en anglès)^[24].

S'ha escollit aquest llenguatge perquè es fa servir molt en diferents branques d'enginyeria, entre elles l'enginyeria química. Això fa possible trobar classes útils pel nostre projecte i no haver de desenvolupar absolutament tot el codi.

En la part d'anàlisis d'aquest document s'indiquen breument els diferents conceptes que formen el conjunt dels elements desenvolupats en el projecte. Així com els atributs de cada classe i una descripció breu de la funcionalitat de les seves funcions. Més endavant, en la secció del document anomenada implementació, es detallen totes les funcions incloses dintre dels conceptes del projecte.

3.2. Model conceptual

El model conceptual o domini permet identificar els conceptes més importants del projecte (informació del sistema polimèric) així com les relacions entre les diferents classes que formen part del codi. Els conceptes estan relacionats per les classes i les relacions entre classes estan representades per les seves associacions.

S'ha escollit el llenguatge unificat de modelatge (UML)^[25] per representar el model conceptual perquè és el més habitual en enginyeria de software. Està suportat per l'OMG (Object Management Grup).^[26] L'UML o Unified Modeling Language, Llenguatge de Modelatge Unificat, és un llenguatge gràfic per visualitzar, especificar, construir i documentar un sistema que ofereix un estàndard per descriure un sistema (model), incloent-hi aspectes conceptuals tals com els processos de negoci i funcions del sistema, i aspectes concrets com expressions de llenguatges de programació, esquemes de bases de dades i components reutilitzables.

L'UML ofereix una forma diferent de visualitzar els problemes mitjançant diagrames; i es tracta d'un llenguatge que permet descriure un model d'anàlisi i disseny d'un sistema mitjançant diagrames construïts utilitzant símbols que tenen regles semàntiques (què significa i com interpretar-ho), sintàctiques (com mostrar i combinar els símbols) i pràctiques (com utilitzar els símbols).

Un model UML és una abstracció que captura el coneixement sobre un problema, és a dir, el model extreu els detalls essencials del problema.

Si es representa la metodologia de desenvolupament de software en diferents fases; és en les tres primeres on s'utilitza el UML.

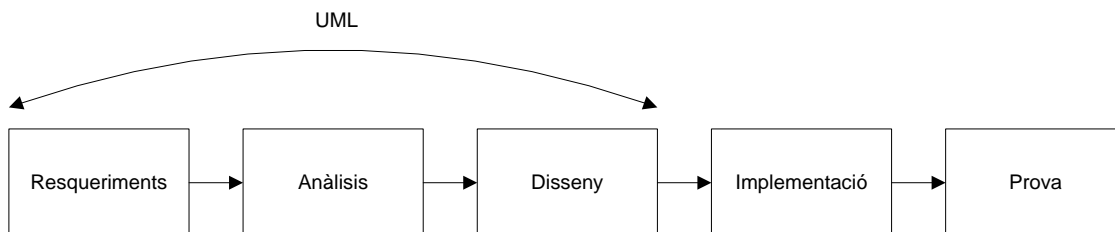


Diagrama 3.2.1. Esquema de metodologia desenvolupament software

En la fase de requeriments s'utilitzen els diagrames de casos d'ús i els d'activitats per descriure el comportament del sistema o funcionalitats i el flux d'activitats involucrades en una funcionalitat determinada del sistema respectivament.

En la següent fase d'anàlisi cal tenir en compte el software i per tant s'utilitzen els diagrames de classes; i per últim en la fase de disseny s'utilitzen diagrames de seqüència i diagrames de comunicació per poder d'escollir els mètodes de cada classe.

Els diagrames de classes descriuen l'estructura d'un sistema mostrant les seves classes, atributs i les relacions entre ells. És on es crea el disseny conceptual de la informació que utilitzarà el sistema, i els components que s'encarregaran del seu funcionament i de la relació entre l'un i l'altre. Les classes principals són les que serveixen per definir quins elements tindrà un sistema. La relació entre les diferents classes ens indica el nombre d'elements que podrà tenir el sistema.

Un model de domini és un artefacte de la disciplina d'anàlisi, construït amb les regles UML durant la fase de concepte, en la tasca de construcció del model de domini, presentant com un o més diagrames de classes i que conté, no conceptes propis d'un sistema de software sinó de la mateixa realitat física.

S'ha dividit el model conceptual en 3 seccions:

- Classes principals.
- Classes d'entrada i sortida.
- Classes de processament.

3.3. Classes principals

Un sistema esta format per molècules, les molècules per residus i els residus per àtoms.

S'han implementat aquests 4 conceptes amb aquestes 4 classes.

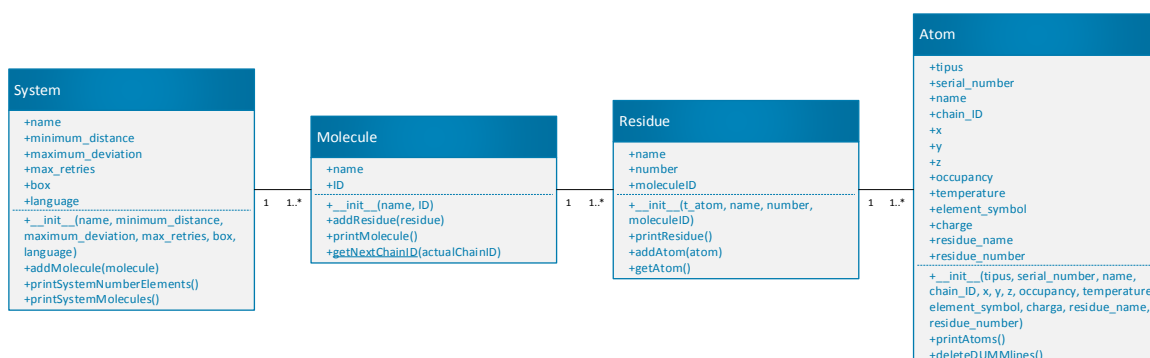


Diagrama 3.3.1. Model conceptual classes principals

El model conceptual de les classes principals ens permet veure que un sistema (matriu polimèrica) podrà tenir una o més molècules (cadena polimèrica), una molècula podrà està formada per un o més residus (monòmer) i un residu podrà està format per un o més àtoms.

Dins de cada classe s'han informat els atributs i les funcions relacionades.

3.3.1. SYSTEM

La classe System és la més important de les classes principals. Ens permet emmagatzemar els principals atributs que formen part d'un sistema, com per exemple el nom del sistema i totes les molècules que formaran el sistema.

Permet afegir noves molècules al sistema mitjançant la funció addMolecule.

3.3.2. MOLECULE

La classe Molecule ens permet realitzar els tractaments relacionats amb les molècules i facilitar la seva identificació gracies als seus atributs, ja sigui consultant el nom o el seu ID.

Permet afegir nous residus dintre de la molècula mitjançant la funció addResidue.

3.3.3. RESIDUE

La classe Residue ens permet realitzar els tractaments dels residus de les molècules. Mitjançant els seus atributs podem identificar el nom del residu, el seu número i a quina molècula pertany.

Permet afegir nous àtoms dintre del residu mitjançant la funció addAtom.

3.3.4. ATOM

La classe Atom ens permet emmagatzemar tota la informació relativa a un àtom, com per exemple el seu nom i el seu tipus o saber a quin residu pertany.

3.4. Classes d'entrada i sortida

En aquest apartat es detallen les classes del sistema que s'utilitzen per realitzar l'Input / Output; és a dir per a la lectura de dades a processar i l'escriptura dels resultats.

Com es mostra en el diagrama 3.4.1 la classe principal "FILES" és una classe genèrica de la qual s'especialitzen dues subclasses: "FILEPDB" i "FILEPREPI", en funció del tipus de fitxer que s'està tractant, sigui per lectura o escriptura.

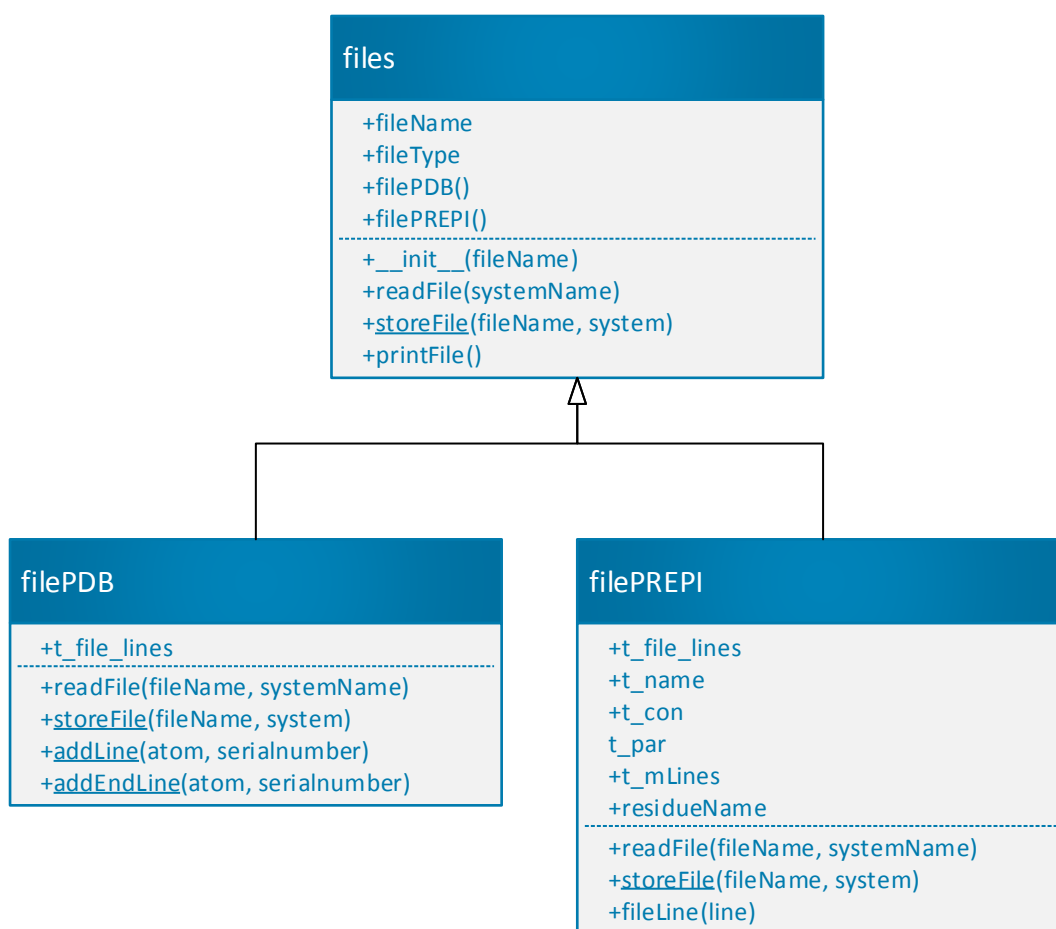


Diagrama 3.4.1. Model conceptual classes input /output

3.4.1. Fitxers d'entrada

La construcció del sistema es realitzarà a partir de residus o monòmers que cal disposar com a fitxers d'entrada en format PREPI; i les indicacions del fitxer de càrrega sobre com s'ha de configurar el nou sistema.

3.4.1.1. Fitxers de càrrega

Per aconseguir crear correctament el fitxer PDB caldrà indicar obligatòriament una sèrie de paràmetres. En cas d'especificar incorrectament aquest fitxer, el programa mostrarà un missatge d'error i no procedirà a la creació del fitxer PDB.

A continuació es llisten i expliquen els paràmetres que cal informar en el fitxer:

- **Nom del fitxer PD:** Indicarà el nom del fitxer PDB que es crearà si no es detecta cap error.
- **Mides de la caixa periòdica 3D:** Cal informar les mesures de la caixa periòdica on es crearà el sistema sempre en Àngstroms (Å) i separats per comes.
- **Coordenades periòdiques:** Indicador per especificar si el sistema creat en el PDB resultant ha de respectar les condicions periòdiques de la caixa.
- **Distància mínima de col·lisió:** És la mínima distància permesa entre dos àtoms no enllaçats del sistema. Aquesta distància sempre serà en Àngstroms (Å).
- **Re-intents:** Número de re-intents màxim que realitzarà el programa per intentar recol·locar un monòmer nou en el sistema sense que es detecti cap col·lisió, és a dir, mantenint el criteri de distància mínima.
- **Desviació màxima:** % de desviació dels paràmetres estructurals dels monòmers que cal tenir en compte per reubicar els nous residus. Quan es detecta una col·lisió s'introdueix una pertorbació aleatòria en les distàncies i angles d'enllaç de cada àtom del monòmer a introduir en el sistema per tal d'evitar-la. Per exemple en cas de tenir una desviació màxima de 5, es considerarà un +/-5% i per tant es calcularà entre el 0,95 i l'1,05 de desviació estructural.
- **Nombre de molècules:** Nombre de molècules que ha de tenir el sistema.
- **Ubicació dels fitxers PREPI:** Nom de la carpeta on es troben tots els fitxers amb format PREPI necessaris per crear el sistema. El nom dels fitxers s'utilitzarà en el fitxer de càrrega per fer referència a cada un dels monòmers necessaris per crear el sistema.
- **Descripció de la composició de les molècules del sistema:** Al final del fitxer cal informar de la composició de cada molècula. S'han d'especificar tantes files com molècules ha de tenir el sistema.

A continuació es mostra una imatge amb un exemple d'un fitxer de càrrega.

```

system_name, PEH-60PE-PET2
box, Lx 75 A, Ly 50 A, Lz 50 A
periodical_coordinates, Y
minimum_distance, 0.1 A
retries, 10
maximum_deviation, 5
molecules_number, 6
path_prepi, prepi
PEH, 60 PE, PET
PEH, 60 PE, PET
PEH, 60 PE, PET
PEH, 60 PE, PET
PEH, 60 PE, PET
PEH, 60 PE, PET

```

Imatge 3.6.1 Imatge exemple de fitxer càrrega

3.4.1.2. Fitxers PREPI

Els fitxers en format PREPI són els que normalment s'utilitzen per a descriure l'estructura de residus peptídics (amb coordenades internes) en el programa **AMBER** ^[27] a l'hora de construir proteïnes. En cada fitxer PREPI es troba un residu o monòmer, que és la unitat molecular bàsica del programa per acabar creant els sistemes.

A continuació es mostra un exemple de fitxer PHE.prepi amb format PREPI de l'aminoàcid fenil alanina, amb les especificacions de cada una de les columnes que componen el fitxer i perquè s'utilitzen en el programa:

```

0 0 1
PHENYLALANINE PREP INPUT EXAMPLE (title)
PHE
PHE INT 1
CORRECT OMIT DU BEG
0.0
1 DUMM DU M 0 -1 -2 0.0000 0.0000 0.0000 0.000
2 DUMM DU M 1 0 -1 1.4490 0.0000 0.0000 0.000
3 DUMM DU M 2 1 0 1.5220 111.1000 0.0000 0.000
4 N N M 3 2 1 1.3350 116.6000 180.0000 -0.5200
5 HN H E 4 3 2 1.0100 119.8000 0.0000 0.2480
6 CA CH M 4 3 2 1.4490 121.9000 180.0000 0.2140
7 CB C2 S 6 4 3 1.5250 111.1000 60.0000 0.0380
8 CG CA S 7 6 4 1.5100 115.0000 180.0000 0.0110
9 CD1 CD S 8 7 6 1.4000 120.0000 180.0000 -0.0110
10 CE1 CD S 9 8 7 1.4000 120.0000 180.0000 0.0040
11 CZ CD S 10 9 8 1.4000 120.0000 0.0000 -0.0030
12 CE2 CD S 11 10 9 1.4000 120.0000 0.0000 0.0040
13 CD2 CD E 12 11 10 1.4000 120.0000 0.0000 -0.0110
14 C C M 6 4 3 1.5220 111.1000 180.0000 0.5260
15 O O E 14 6 4 1.2290 120.5000 0.0000 -0.5000

IMPROPER
-M CA N HN
CA +M C O
CB CA N C

LOOP
CG CD2

DONE
STOP

```

Figura 3.7.1 Imatge exemple de fitxer PREPI del residu/monòmer PHE

- Columna 1 – Comptador de les posicions del fitxer PREPI.
- Columna 2 – Un nom atòmic exclusiu de l'àtom per a cada residu o monòmer.
- Columna 3 – Símbol del tipus d'àtom que defineix el seu comportament o interaccions atòmiques amb la resta d'àtoms del sistema. S'utilitza per assignar els paràmetres del camp de força (force-field). No s'utilitza en el programa SuSi.
- Columna 4 – El tipus topològic (símbol d'arbre) per l'àtom I (M, S, B, E, o 3)

Els àtoms en un residu es classifiquen en cinc tipus topològics: "Main", "Side", "Branch", "3", "4", "5", "6" i "Final". Es denoten com M, S, B, 3 4 5 6 i E respectivament.

Els àtoms principals descriuen el "camí" principal a través del residu, començant per la connexió amb el residu anterior i acabant en la connexió amb el següent residu. El programa connectarà l'últim àtom principal d'un residu al primer àtom principal del següent residu de la molècula. Si només hi ha un residu en una molècula, els àtoms principals solen ser la cadena continua més llarga. Els àtoms principals (M) poden tenir 1, 2, 3 o 4 àtoms connectats a ells.

Qualsevol àtom que no sigui un àtom principal es descriu per un dels altres tipus topològics: "E", "S", "B", "3", "4", "5" o "6". Un àtom "E" només té una connexió amb altres àtoms, de manera que és un "extrem sense sortida" per a qualsevol branca de qualsevol altre tipus d'àtom. Un àtom "S" ha de tenir un total de dues connexions amb altres àtoms, un àtom "B" ha de tenir un total de tres connexions, i un àtom "3" en realitat té un total de quatre connexions; el mateix s'aplica a "4" "5" i "6".

- Columna 5 – NA(I) El número de l'àtom al qual està connectat l'àtom "I".
- Columna 6 – NB(I) El número de l'àtom en què l'àtom "I" fa un angle juntament amb NA(I).
- Columna 7 – NC(I) El nombre de l'àtom en què l'àtom "I" fa un díedre juntament amb NA (I) i NB (I).
- Columna 8 – R(I) Aquesta és la longitud de connexió entre els àtoms "I" i NA (I).
- Columna 9 – THETA(I) és l'angle de connexió entre l'àtom NB (I), NA (I) i "I".
- Columna 10 – PHI(I) és l'angle diedre entre NC (I), NB (I), NA (I) i "I".
- Columna 11 – CHRG (I) La càrrega atòmica parcial de l'àtom "I".

Cal tenir en compte que els fitxers PREPI sempre esperen tres àtoms ficticis per començar.

Els àtoms ficticis precedeixen els àtoms reals del residu. Aquests àtoms només s'utilitzen per definir els eixos espacials del residu. Els tres àtoms ficticis han de tenir el tipus topològic "M", i se'ls ha d'assignar un tipus d'àtom de camp de força que els defineix com a àtoms ficticis.

El programa desenvolupat en la classe filePREPI utilitza el contingut de les columnes 5, 6, 7, 8, 9 i 10 per enllaçar els residus.

3.4.2. Fitxers de sortida

El resultat de l'execució del programa serà un sistema (matriu polimèrica) en un fitxer en format PDB o bé un missatge amb el resultat de l'execució.

3.4.2.1. Fitxer format PDB

El format del Banc de dades de proteïnes (Protein Data Bank, PDB) ^[28] proporciona una representació estàndard per a les dades de l'estructura macromolecular derivades de la difracció de raigs X i els estudis de RMN.

Aquesta representació es va crear a la dècada de 1970 i s'ha escrit una gran quantitat de programari que l'utilitza. En concret es va inventar el 1976 com un arxiu que permetria als investigadors intercanviar coordenades de proteïnes a través d'un sistema de bases de dades. El seu format original estava limitat a 80 columnes, que es basava en l'amplada de les targetes de perforació de l'ordinador que abans es feien servir per intercanviar les coordenades.

El format de fitxer del Banc de dades de proteïna (PDB) és un format d'arxiu textual que descriu les estructures tridimensionals de les molècules que es troben al Banc de dades de proteïnes. El format de PDB en conseqüència permet la descripció i anotació d'estructures de proteïnes i àcids nucleics, incloses les coordenades atòmiques, assignacions d'estructures secundàries i connectivitat atòmica.

El fitxer es divideix en diverses seccions:

- Registres HEADER, TITLE i AUTHOR: Proporcionar informació sobre els investigadors que han definit l'estructura.
- REMARK: En aquesta secció poden incloure's anotacions, però també altra informació estandarditzada.
- Registres de SEQRES: Dóna les seqüències de les tres cadenes peptídiques (anomenades A, B i C).
- Registres ATOM: És on es descriuen les coordenades dels àtoms que formen part de la proteïna.
- Registres HETATM: Descriuen les coordenades d'heteroàtoms, és a dir, aquells àtoms que no formen part de la molècula de proteïnes.

A continuació es mostra un fitxer PDB amb les seves seccions:

```

HEADER    EXTRACELLULAR MATRIX                22-JAN-98   1A3I
TITLE     X-RAY CRYSTALLOGRAPHIC DETERMINATION OF A COLLAGEN-LIKE
TITLE     2 PEPTIDE WITH THE REPEATING SEQUENCE (PRO-PRO-GLY)
...
EXPDTA    X-RAY DIFFRACTION
AUTHOR    R.Z.KRAMER,L.VITAGLIANO,J.BELLA,R.BERISIO,L.MAZZARELLA,
AUTHOR    2 B.BRODSKY,A.ZAGARI,H.M.BERMAN
...
REMARK 350 BIOMOLECULE: 1
REMARK 350 APPLY THE FOLLOWING TO CHAINS: A, B, C
REMARK 350   BIOMT1   1  1.000000  0.000000  0.000000      0.00000
REMARK 350   BIOMT2   1  0.000000  1.000000  0.000000      0.00000
...
SEQRES   1 A    9  PRO PRO GLY PRO PRO GLY PRO PRO GLY
SEQRES   1 B    6  PRO PRO GLY PRO PRO GLY
SEQRES   1 C    6  PRO PRO GLY PRO PRO GLY
...
ATOM      1  N    PRO A   1      8.316  21.206  21.530  1.00 17.44      N
ATOM      2  CA   PRO A   1      7.608  20.729  20.336  1.00 17.44      C
ATOM      3  C    PRO A   1      8.487  20.707  19.092  1.00 17.44      C
ATOM      4  O    PRO A   1      9.466  21.457  19.005  1.00 17.44      O
ATOM      5  CB   PRO A   1      6.460  21.723  20.211  1.00 22.26      C
...
HETATM   130  C    ACY   401      3.682  22.541  11.236  1.00 21.19      C
HETATM   131  O    ACY   401      2.807  23.097  10.553  1.00 21.19      O
HETATM   132  OXT  ACY   401      4.306  23.101  12.291  1.00 21.19      O
...

```

Figura 3.8.1 Imatge estructura de fitxer PDB

El fitxer PDB de sortida del programa sols haurà d'escriure la secció de registres ATOM que és la necessària per poder interpretar el sistema. A continuació es llisten cada una de les columnes del fitxer PDB en l'estructura ATOM:

Tipus	Columna	Descripció	Justificació	Tipus de dada
ATOM	1-4	"ATOM"		caràcter
	7-11	Nombre de sèrie de l'àtom	dreta	enter
	13-16	Nom de l'àtom	esquerra	caràcter
	17	Indicador d'ubicació alternatiu		caràcter
	18-20	Nom del residu	dreta	caràcter

22	Identificador de la cadena		caràcter
23-26	Nombre seqüencial del residu	dreta	enter
27	Codi per a inserir residus		caràcter
31-38	X coordinada ortogonal Å	dreta	real (8.3)
39-46	Y coordinada ortogonal Å	dreta	real (8.3)
47-54	Z coordinada ortogonal Å	dreta	real (8.3)
55-60	Occupació	dreta	real (6.2)
61-66	Factor de temperatura	dreta	real (6.2)
73-76	Identificació del segment	dreta	caràcter
77-78	Símbol de l'element	dreta	caràcter
79-80	Càrrega		caràcter

Figura 3.8.2 Taula amb columnes de la secció àtom

L'aspecte del fitxer PDB resultant tindrà l'aspecte de la següent imatge:

```

ATOM      1  N   ALA A  1   -1.727   2.046   0.000   1.00   0.00           N
ATOM      2  H   ALA A  1   -1.759   1.037   0.000   1.00   0.00           H
ATOM      3  CA  ALA A  1   -2.980   2.773  -0.000   1.00   0.00           C
ATOM      4  HA  ALA A  1   -3.037   3.400   0.890   1.00   0.00           H
ATOM      5  CB  ALA A  1   -3.098   3.664  -1.232   1.00   0.00           C
ATOM      6  HB1 ALA A  1   -3.545   3.553  -2.220   1.00   0.00           H
ATOM      7  HB2 ALA A  1   -3.727   4.312  -0.621   1.00   0.00           H
ATOM      8  HB3 ALA A  1   -2.107   4.108  -1.330   1.00   0.00           H
ATOM      9  C   ALA A  1   -4.167   1.820   0.000   1.00   0.00           C
ATOM     10  O   ALA A  1   -5.316   1.385   0.000   1.00   0.00           O
TER       11          ALA A  1

```

Figura 3.4.2.3 Imatge fitxer PDB generat

En la classe filePREPI el programa genera el fitxer PDB informant en files cada una de les columnes necessàries:

- ATOM: Per defecte s'informa "ATOM" a l'inici de cada fila



- Nombre de sèrie de l'àtom: Comptador de cada una de les files iniciant amb 1
- Nom de l'àtom: extret del fitxer PREPI
- Nom del residu: Extret del nom del fitxer PREPI i el fitxer de càrrega.
- ID cadena: Numeració per la molècula seguint l'ordenació de l'abecedari.
- Codi numèric per identificar els residus.
- Coordenada X/Y/Z: Coordenades ortogonals amb la posició de l'àtom obtinguda a partir de les coordenades esfèriques del fitxer PREPI i la col·locació d'aquest en el sistema.
- Ocupació: Per defecte es posarà "1.00".
- Factor de la temperatura: Per defecte sempre es posarà "0.00".
- Símbol: Mateix valor que en el nombre de sèrie.
- Càrrega: Per defecte es posarà "0.00".

3.4.2.2. Missatges

Si el procés no s'executa correctament, no es crea el fitxer PDB sinó que mostra l'error que ha fet aturar la generació de les coordenades finals del sistema.

Els errors estan numerats i es troben enumerats en la classe MESSAGE en anglès; però es podrien fàcilment configurar en altres idiomes.

3.4.3. Contingut de les classes

3.4.3.1. FILES

Es la classe principal pel tractament de fitxers. En funció del tipus de fitxer, ja sigui PDB o PREPI el programa determinarà si s'han de fer servir mètodes de la classe FilePDB o FilePREPI.

3.4.3.2. FILEPDB

Aquesta classe permet realitzar els tractaments sobre els fitxers PDB, com per exemple llegir el fitxer o emmagatzemar un nou fitxer PDB.

La seva funció principal storeFile i permet la creació d'un fitxer PDB.

3.4.3.3. FILEPREPI

Aquesta classe es fa servir per realitzar el tractament dels fitxers PREPI que es fan servir al principi del programa on cada fitxer PREPI té emmagatzemada les dades a llegir indicades pel fitxer de carrega.

La funció que més és fa servir es readFile, que ens permet llegir el fitxer PREPI.

3.4.3.4. MESSAGES

Aquesta classe permetrà el tractament dels missatges en diferents idiomes. En ella s'han donat d'alta tots els missatges d'error previstos en l'execució del programa.

Els missatges es troben numerats; i tot i que actualment s'han creat en anglès, s'ha preparat la funció de forma que posteriorment es puguin afegir altres llenguatges. En tal cas es podrà realitzar el canvi adaptant la funció `getTextMessage` inclosa dintre d'aquesta classe.



Figura 3.4.4.1 Classe de missatges

A continuació es llisten els missatges donats d'alta en la classe numerats; i es pot observar que alguns utilitzen variables que es substitueixen pel paràmetre "&":

```

'001' : 'File can't have less than 8 lines',
'002' : 'Collisions could not be avoided for this molecule',
'003' : 'system_name not found in file',
'004' : 'box dimension not found',
'005' : 'periodical_coordinates not found',
'006' : 'minimum_distance not found',
'007' : 'retries not found in file',
'008' : 'maximum_deviation not found in file',
'009' : 'molecules_number not found in file',
'010' : 'PATH of prepi files not found in file',
'011' : 'Error in X dimension unit',
'012' : 'Error in Y dimension unit',
'013' : '',
'014' : 'Error in Z dimension unit',
'015' : 'Error in minimum distance unit',
'016' : 'System name not found in file',
'017' : 'Error in value of periodical coordinates. Possible values are Y or N',
'018' : 'Error in minimum distance',
'019' : 'Molecules number not found in file',
'020' : 'PATH of prepi files not found in file',
'021' : 'Molecules number in file different than molecules number found',
'022' : 'Error in box',
'023' : 'Could not read file: &',
'024' : 'No errors found',
'025' : 'Could not create file: &',

```

Figura 3.4.4.2 Taula amb el contingut dels missatges

3.5. Classes de processament

Les classes de processament permeten relacionar totes les classes del projecte.

3.5.1. BUILDER

Es tracta de la classe principal de l'aplicació. És l'encarregada de llegir el fitxer d'entrada, els fitxers PREPI i de realitzar la crida a la resta de funcions d'altres classes per crear el fitxer PDB.

A continuació es detalla el processament de cada un dels passos del programa que s'ha definit el diagrama del 3.5 d'aquest apartat:

PAS 1: Abans de fer començar el procés de la creació del sistema, cal comprovar si les dades introduïdes al fitxer d'entrada son correctes. Si no son correctes aleshores emmagatzemarà els errors i anirà al pas 3 per poder mostrar els errors per pantalla.

PAS 2: Dintre del fitxer de carrega s'indica quins residus ha de tractar el programa per aconseguir la creació del sistema Però es possible, que per error, algun d'aquests fitxers no es trobi dins del directori indicat al fitxer de carrega, però això en aquest pas es controla si tots els fitxers amb les dades dels residus (fitxers PREPI) estan al directori indicat. En cas de trobar algun error, el va emmagatzemant perquè posteriorment al pas 3 el pugui mostrar per pantalla.

PAS 3: Si ha detectat algun error en l'execució dels passos anteriors aleshores no continuarà amb el pas 6, sinó que passarà als passos 4 i 5 per mostrar els errors per pantalla.

PASSOS 4 i 5: Si ha arribat a aquest pas es perquè s'ha produït algun tipus d'error, per tant cal veure quins errors s'han produït i obtenir el texts dels error en el llenguatge desitjat. Un cop s'ha obtingut tota la informació podrà mostrar per pantalla els missatges d'error.

PASSOS 6 i 7: Si ha arribat a aquest pas és perquè sabem segur que tots els fitxers amb residus (PREPI) indicats al fitxer de carrega estan dintre del directori de fitxers PREPI. Per tant el que fa és anar tractant tots els fitxers un per un, llegint el seu contingut i emmagatzemant la seva informació per poder fer-la servir més endavant.

PAS 8: A partir d'aquest pas es on es comença el procés d'intentar afegir noves molècules al sistema.

La primera vegada que es fa aquesta comprovació no farà res, ja que no s'haurà superat el nombre de intents, per tant anirà directament al pas 9, però a partir de la segona vegada si és possible que es compleixi aquest criteri.

Cada cop que passi per aquest pas comprovarà si s'ha superat el nombre d'intents d'afegir la molècula actual dins el sistema, si aquest número d'intents és superior a l'indicat al fitxer de càrrega, no continuarà el procés i anirà al pas 4 per mostrar l'error.

PAS 9: Per poder afegir una nova molècula al sistema, es comprova que cap dels àtoms dels seus residus es pugui considerar que estan en col·lisió. Considerarem que un àtom està en col·lisió si la seva distància amb els altres àtoms es inferior a la distància indicada en el fitxer de carrega.

Si no es detecta cap col·lisió incorporarà aquesta nova molècula, amb els seus residus i àtoms, al sistema i anirà al pas 11 i després al 12.

Si es detecta una col·lisió no afegirà aquesta molècula al sistema i anirà al pas 11 i després al 10.

PAS 10: Arribarà a aquest pas només si ha detectat que hi ha una col·lisió. En aquest cas el que farà serà intentar col·locar la molècula d'una altra manera en la que no es produeixi cap col·lisió. En aquest pas augmentarà en un el nombre de intents que s'han fet per aquesta molècula, perquè després quan s'executi el pas 8 pugui detectar si s'ha superat el nombre màxim de intents.

PAS 11: Si detecta que s'ha produït una col·lisió anirà al pas 10, sinó anirà al pas 11.

PAS 13: En aquest pas s'obtindrà la propera molècula a tractar per que pugui ser tractada al pas 9.

PAS 14: Quan arribi a aquest pas voldrà dir que ja s'han tractat totes les molècules. El que fa ara és comprovar si el procés s'estava executant en mode test (anar al pas 15) o en mode normal (anar al pas 16).

PAS 15: En mode test vol dir que no s'ha de crear el fitxer amb les dades del sistema (fitxer PDB), sinó que s'ha de mostrar informació del sistema per pantalla.

PAS 16: Si s'està executant en mode normal, vol dir que cal crear el fitxer amb les dades del sistema (fitxer PDB), però abans d'això cal saber si volem un sistema amb coordenades periòdiques (pas 17) o sense coordenades periòdiques (pas 18).

PAS17: S'ha demanat un fitxer sense coordenades periòdiques, per tant es genera un fitxer en aquestes condicions al pas 19.

PAS 18: S'ha demanat un fitxer amb coordenades periòdiques, per tant es genera un fitxer en aquestes condicions al pas 19.

PAS 19: S'arriba al fi del procés, on es procedeix a crear el fitxer amb les dades del sistema (fitxer PDB).

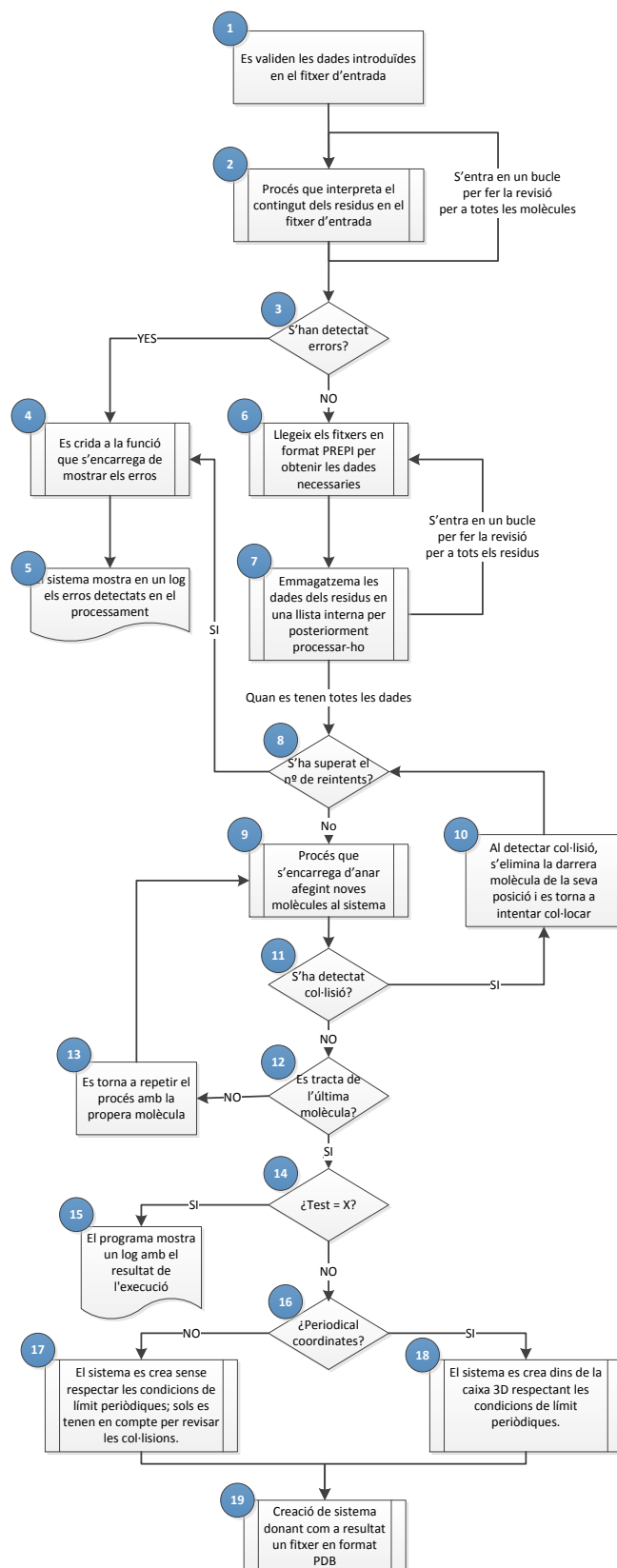


Diagrama 3.5.1.1 Esquema funcionalitat Builder

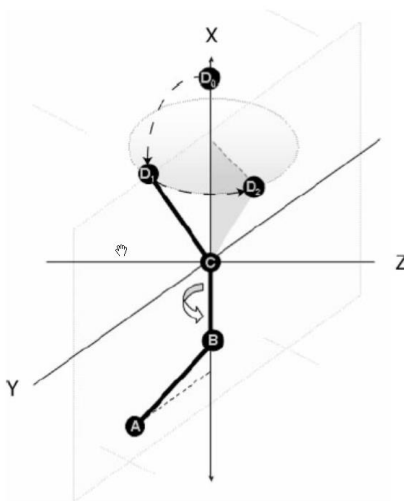
3.5.2. INT2CAR_M

Classe formada per diferents funcions que realitzen càlculs matemàtics per poder realitzar el tractament de les distàncies i dels angles.

Com moltes altres aplicacions en aquest programa s'ha requerit d'un mètode per traduir els angles i longituds d'enllaç dels àtoms a coordenades cartesianes. Es tracta d'una tasca senzilla però fonamental en la modelització de polímers. Per aquest motiu després d'analitzar diversos estudis sobre el millor mètode a seguir; es va utilitzar codi font que es recomanava en l'article "Conversió pràctica des de l'espai de torsió a l'espai cartesià per a la síntesi de proteïnes de sílice"^[29]

En aquest estudi es va tenir en compte que els polímers consisteixen en sèries d'àtoms que estan ocupats per una estructura tridimensional i lligats consecutivament. Aquesta estructura pot estar representada per les coordenades cartesianes dels àtoms (base natural que descriu els camps de força física i les densitats dielèctriques) o, alternativament, com a longituds i angles d'enllaç(descripció química de l'enllaç covalent).

La conclusió de l'estudi és que el mètode "SN-NeRF buildup" és el millor per al càlcul de coordenades cartesianes a partir dels paràmetres espacials de torsió; ja que la resta de algoritmes que van analitzar eren susceptibles a un error d'arrodoniment. Finalment, es va establir que l'enfocament més pràctic i eficaç per calcular les coordenades cartesianes era passar la proteïna de forma sistemàtica; és a dir, les posicions atòmiques es van determinant a mesura que s'avança per la columna vertebral de la proteïna i després aquesta s'utilitza en càlculs posteriors.



Imatge 3.5.3.1 Imatge de transformació de coordenades cartesianes^[29]

4. Implementació

A continuació es descriurà amb més detall les funcions contingudes dins de les classes, tant els seus paràmetres d'entrada i sortida així com el funcionament de cada funció.

4.1. BUILDER

La classe BUILDER és l'encarregada de llegir el fitxer d'entrada, els fitxers PREPI i de realitzar la crida a la resta de funcions d'altres classes per crear el fitxer PDB.

builder

```
+system
+system2
+verbosity
+logFile
-----
+__init__(fileName, verb, test)
+getAndCheckFileValues(t_lines)
+checkRepeatedResidues(t_residues)
-__addNewMolecule(t_molecule, moleculeNumber, moleculeID, t_filePREPI, t_log)
-__processResidue(numResidue, residueName, numRetry, moleculeNumber, t_filePREPI, t_xyz, t_last3m, t_log)
-__addNewResidue(t_xyz, filePREPI, numResidue, moleculeID, t_log)
+convertxyz(t_xyz)
+convertxyz2(t_xyz)
+convertToAtoms(t_xyz, t_atomName, residueName, residueNumber, chainID)
+printErrors(t_errors, language)
+getLast3M(nLines, t_xyz, numResidue, t_oldLast3m)
+PeriodicBoundaryCondition(box, posX, posY, posZ)
-__checkCollisionInSystem(moleculeID, atom, t_log)
-__checkCollisionInSystem2(posX, posY, posZ)
-__getRandomPositionWithoutCollision()
+getRandomThetaAndPhi(t_par, maximumDeviation)
+getSystem()
```

Diagrama 4.1.1. Model conceptual classe Builder

A continuació es mostrarà el diagrama en forma d'esquema de funcionament de la classe Builder; i en els punts posteriors es detallaran les funcions de la classe anomenant els seus paràmetres d'entrada i sortida juntament amb la descripció de la funció i el codi utilitzat.

4.1.1. Diagrama del funcionament de la classe Builder.

En primer lloc es mostra l'esquema de la classe principal i l'orde de les funcions a les quals es van cridant al llarg del procés; i posteriorment es mostren els diagrames de flux de la classe "Builder" i de la funció "addNewMolecule" i "processResidue".

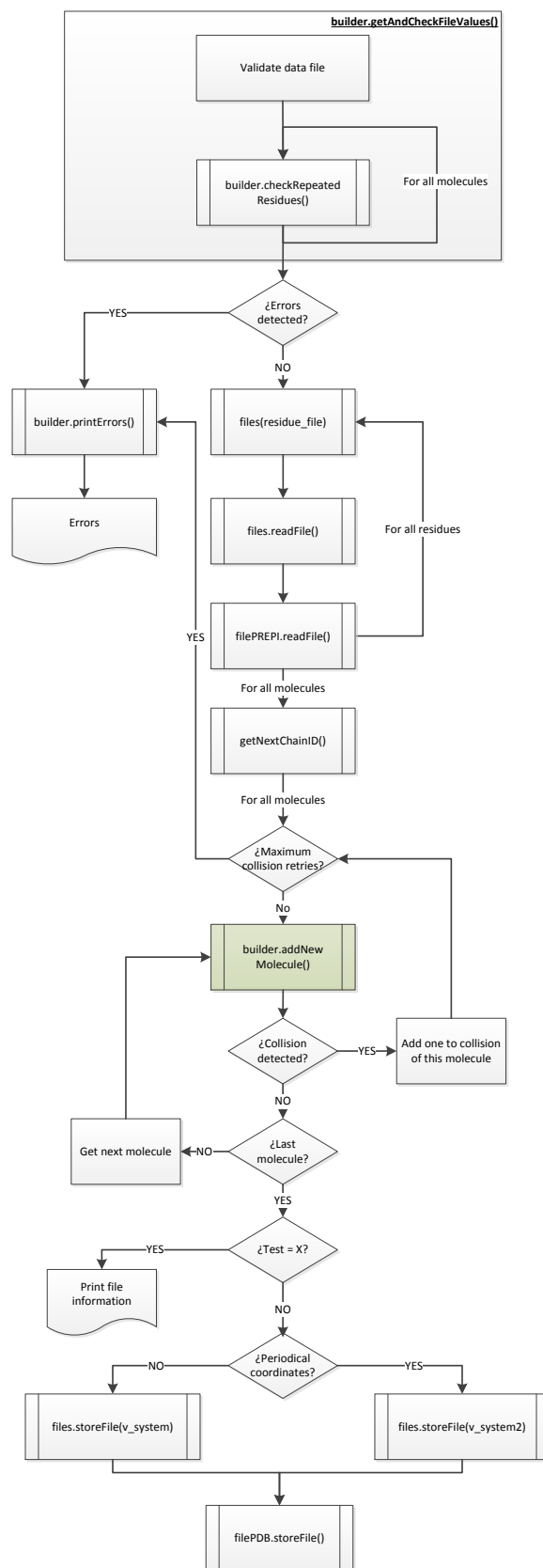


Diagrama 4.1.1. Diagrama de flux de la classe builder

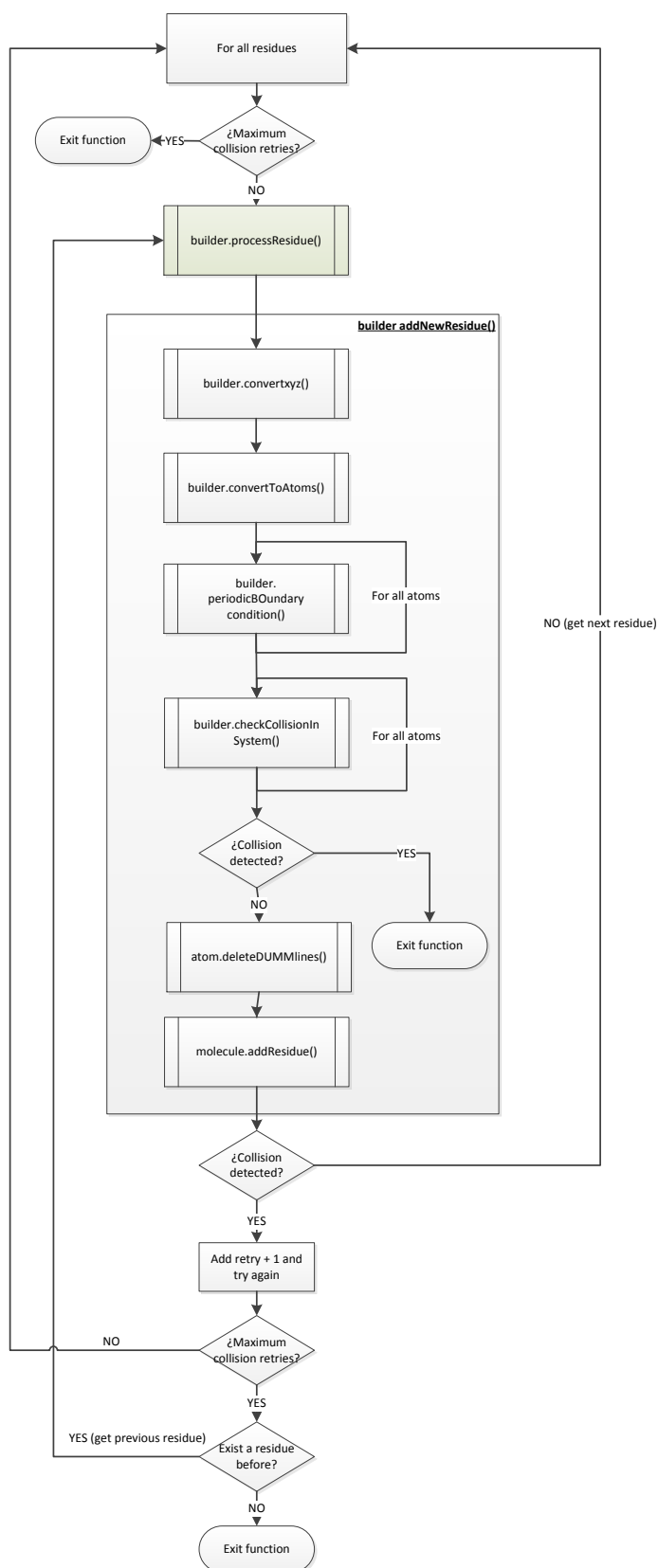


Diagrama 4.1.2. Funció addNewMolecule

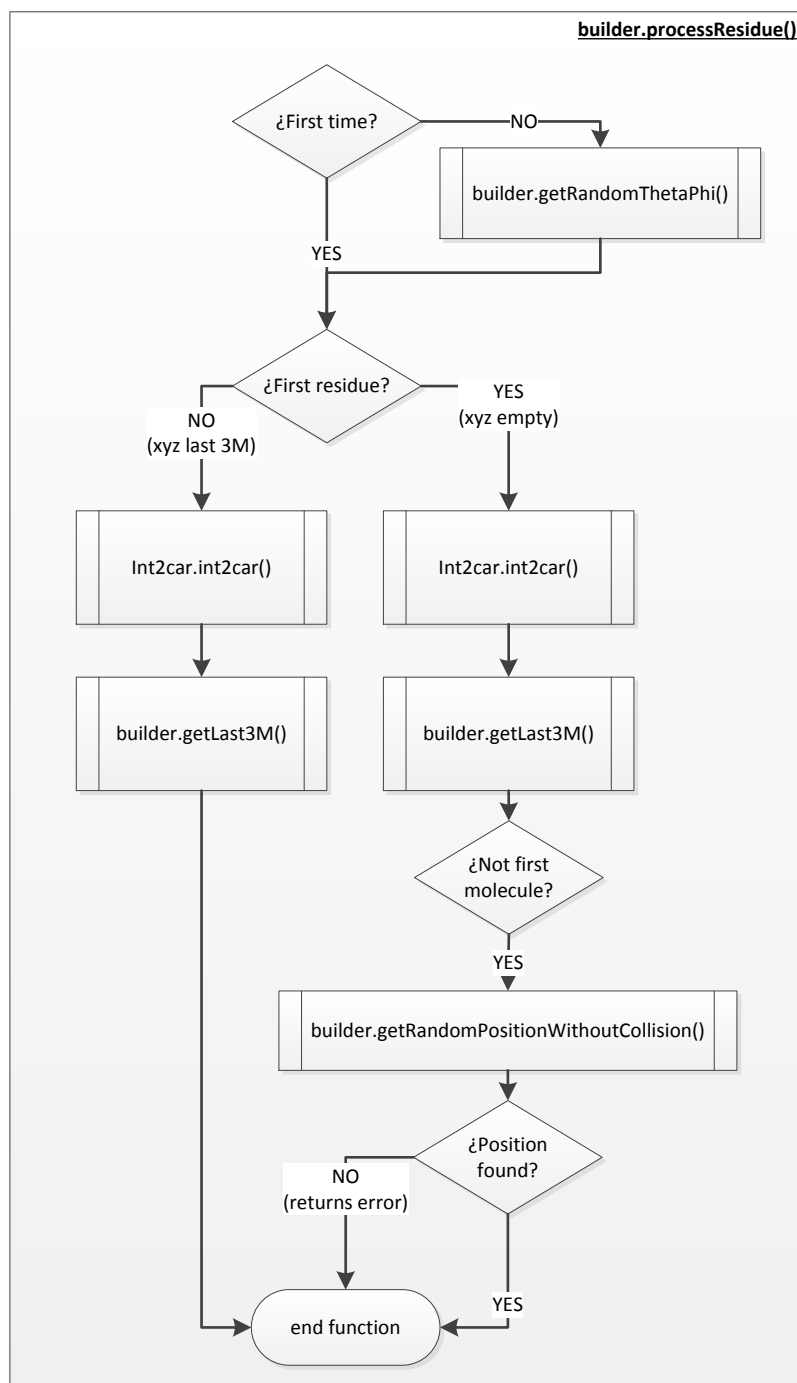


Diagrama 4.1.3. Diagrama de flux de la funció processResidue()

4.1.2. Funció __init__

Paràmetres d'entrada	Descripció
imFileName	Nom del fitxer.
verb	= 'X' en finalitzar el procés de creació del fitxer mostrarà per pantalla informació del que ha succeït.
imTest	Mode d'execució de la funció. = 'X' mode test activat. = ' ' mode test no activat. És un paràmetre opcional, és a dir, si no es vol fer servir el mode test, no caldrà informar-lo.

Taula 4.1.2.1. Paràmetres d'entrada

Descripció:

És la funció principal de tot el procés de creació del fitxer PDB. S'encarrega de realitzar la crida a la resta de funcions de la classe BUILDER i a cridar funcions d'altres classes.

Procés que realitza la funció:

- Activa el log de procés si el paràmetre d'entrada verb té el valor 'X'.
- Realitza algunes comprovacions per veure si el fitxer d'entrada té les dades correctament informades:
 - Comprova si el fitxer d'entrada existeix realment, sinó retorna error i no continua el procés.
 - Comprova si el fitxer té com a mínim 9 línies, que es el contingut mínim que ha de tenir, sinó retorna error i no continua el procés.
 - Crida a la funció getAndCheckFileValues per realitzar la resta de validacions del fitxer. Aquesta funció s'explica amb més detall una mica més endavant.
 - Si detecta errors, els mostrarà per pantalla amb la funció printErrors i no continua el procés.
- Realitza el tractament de tots els residus indicats dintre del fitxer de càrrega:
 - Llegeix totes les molècules indicades al fitxer de càrrega.
 - Comprova que tots els residus tinguin el fitxer PREPI corresponent dins la ruta indicada al fitxer de càrrega on trobar els PREPIs (path_prepi). En cas de no trobar algun fitxer, mostrarà un missatge d'error i no continuarà el procés.
 - Llegeix el contingut dels fitxers PREPI i va emmagatzemant la informació dins de la variable tipus llista anomenada t_filePREPI.
- Crea 2 variables de tipus SYSTEM: la primera SYSTEM on s'apliquen condicions periòdiques de contorn (PBC, Periodic Boundary Conditions) , i la segona SYSTEM2 sense aplicar-les.

5. Llegeix totes les molècules que s'hauran d'afegir al sistema i determina quin serà el seu ID mitjançant la funció getNextChainID.
6. Comença el tractament de totes les molècules indicades al fitxer d'entrada:
 - a. Comprova si ja s'ha realitzat el nombre màxim d'intents per col·locar la molècula dins el sistema. En cas que sigui així finalitza el procés, en cas contrari continua amb el següent punt.
 - b. Crida a la funció __addNewMolecule per intentar afegir la molècula dins el sistema. En cas de detectar una col·lisió, anirà al punt anterior (punt a).
7. Un cop tractades totes les molècules podran succeir dos coses:
 - a. Si s'està executant en mode test, aleshores mostrarà per pantalla informació relativa al fitxer llegit per poder comprovar si la funció tractarà correctament les dades introduïdes.
 - b. Si no s'està executant en mode test, realitzarà la creació del fitxer PDB mitjançant el mètode storeFile de la classe FILES.

4.1.3. Funció getAndCheckFileValues

Paràmetres d'entrada	Descripció
t_lines	Línies del fitxer d'entrada.
Paràmetres de sortida	Descripció
Box	Llista amb les dades de la capsa. box[0]: Dimensió X, box[1]: Unitat de la dimensió X. box[2]: Dimensió Y, box[3]: Unitat de la dimensió Y. box[4]: Dimensió Z, box[5]: Unitat de la dimensió Z.
periodical_coordinates	Determina si en el fitxer que s'ha d'obtenir s'han d'aplicar les condicions periòdiques de contorn (PBC). Y: Aplicar PBC. N: No aplicar PBC.
minimum_distance	Distància mínima que es considerarà que s'ha produït una col·lisió.
maximum_deviation	Valor de la desviació màxima.
system_name	Nom del sistema.
pathPrepi	Ruta on estan emmagatzemats els fitxers PREPI.
t_molecules	Llista de les molècules obtingudes del fitxer.
t_errors	Llista amb els errors detectats.

Taula 4.1.3.1. Paràmetres d'entrada/sortida

Descripció:

Llegeix les línies del fitxer d'entrada i fa dues coses:

- Comprova que les dades introduïdes al fitxer siguin correctes.

- Si les dades són correctes retorna el valor de les línies del fitxer en diferents paràmetres de sortida.

Les diferents validacions que fa són:

Validació	Missatge d'error
Comprova que el fitxer contingui el nom del sistema.	System name not found in file.
Comprova que el fitxer contingui les dades de la caps. Dimensions i unitats.	Box dimension not found.
Comprova que el fitxer contingui un valor al camp <code>periodical_coordinates</code> .	Periodical coordinates not found.
Comprova que el fitxer contingui el valor de la mínima distància de les col·lisions.	Minimum distance not found.
Comprova que en el fitxer s'hagi informat el valor dels reintents	Retries not found in file
Comprova que el fitxer contingui el valor del "màxim deviation".	Maximum deviation not found in file.
Comprova que el fitxer contingui el número de molècules que ha de contenir el fitxer.	Molecules number not found in file.
Comprova que el fitxer contingui la ruta del directori on estan emmagatzemats els fitxers PREPI.	PATH of prepi files not found in file.
Comprova si la unitat de la dimensió X es correcta.	Error in X dimension unit.
Comprova si la unitat de la dimensió Y es correcta.	Error in Y dimension unit.
Comprova si la unitat de la dimensió Z es correcta.	Error in Z dimension unit.
Comprova si la unitat de la mínima distància de col·lisió es correcte.	Error in minimum distance unit.
Comprova si es correcte el valor de la mínima distància de col·lisió.	Error in minimum distance.
Comprova si existeix el directori on estan els fitxers PREPI.	Could not found PATH
Comprova si el número de molècules informades es el mateix que la variable <code>molecules_number</code> .	Molecules number in file different than molecules number found.

Taula 4.1.3.2. Taula de validacions

4.1.4. Funció `checkRepeatedResidues`

Paràmetres d'entrada	Descripció
<code>t_residues</code>	Llista amb els valors d'una línia de residus. Aquestes línies són les indicades a la part final de la creació del fitxer.
Paràmetres de sortida	Descripció
<code>t_auxResidues</code>	Llista amb els valors de la línia de residus. És possible que sigui el mateix valor que el d'entrada o que s'hagi modificat. En funció si la línia indica que hi ha residus adjacents indicats amb un número. Exemple: 5 ALA.

Taula 4.1.4.1. Paràmetres d'entrada/sortida

Descripció:

Aquesta funció s'ha creat per facilitar la manera d'omplir el fitxer. Possibilita l'opció d'utilitzar una forma compacta d'escriure la composició de les molècules amb el detall dels residus quan es tracta de posicions adjacents del mateix tipus.

Si es tracta la molècula mol1, que té els 5 residus ALA adjacents una darrera l'altra. Al fitxer es pot indicar:

- sense comprimir els residus: mol1, ALA, ALA, ALA, ALA, ALA
- comprimint els residus: mol1, 5 ALA

De manera transparent a l'usuari, la funció converteix el mol1, 5 ALA en mol1, ALA, ALA, ALA, ALA, ALA.

4.1.5. Funció __addNewMolecule

Paràmetres d'entrada	Descripció
it_molecule	Llista amb els residus que tindrà la molècula.
iv_moleculeNumber	Número de la molècula actual.
iv_moleculeID	ID de la molècula actual.
to_filePREPI	Llista amb totes les variables tipus PREPI.
t_log	Llista amb el resultat del procés.
Paràmetres de sortida	Descripció
collision	True si s'ha detectat una col·lisió, False en cas contrari.

Taula 4.1.5.1. Paràmetres d'entrada/sortida

Descripció:

Comença el tractament per intentar afegir una nova molècula dintre del sistema.

Els passos que fa la funció són:

1. Afegeix una nova molècula buida als sistemes System i System2.
2. Llegeix tots els residus indicats a la llista de molècules (paràmetre d'entrada it_molecule) i realitza el següent tractament sobre cada residu:
 - a. Comprova si ja s'ha realitzat el nombre màxim d'intents per col·locar el residu dins el sistema. En cas que sigui així finalitza la funció, en cas contrari continua amb el següent punt.
 - b. Executa la funció __processResidue per realitzar el tractament sobre aquest residu (més endavant està explicada en detall).

- c. Executa la funció `__addNewResidue` per tractar d'afegir aquest residu al sistema. En cas de no poder afegir-lo perquè ha detectat una col·lisió, continua al punt a) amb el mateix residu. En cas de no detectar cap col·lisió continua el tractament del següent residu anant al punt a.
3. Retorna un error si hi ha hagut alguna col·lisió.

4.1.6. Funció `__processResidue`

Paràmetres d'entrada	Descripció
<code>iv_numResidue</code>	Número del residu actual.
<code>iv_residueName</code>	Llista amb els residus que tindrà la molècula.
<code>iv_numRetry</code>	Número d'intents d'afegir el residu sense col·lisió en els seus àtoms.
<code>iv_moleculeNumber</code>	Numero de la molècula actual.
<code>it_filePREPI</code>	Llista amb totes les variables tipus PREPI.
<code>it_xyz</code>	Llista obtinguda amb la funció <code>int2car</code> .
<code>it_last3m</code>	Valors obtinguts amb la funció <code>int2car</code> .
<code>t_log</code>	Llista amb el resultat del procés.
Paràmetres de sortida	Descripció
<code>it_xyz</code>	Valor actualitzat de la llista <code>it_xyz</code> .
<code>v_filePREPI</code>	Fitxer PREPI del residu actual. Obtingut de la llista <code>it_filePREPI</code> .
<code>it_last3m</code>	Valor actualitzat de la llista <code>it_last3m</code> .

Taula 4.1.6.1. Paràmetres d'entrada/sortida

Descripció:

Realitza el tractament per comprovar si aquest nou residu es pot afegir a la molècula o si pel contrari no es pot perquè els àtoms col·lisionen amb altres àtoms del sistema.

Els passos que fa la funció són:

- 1- Comprova si és el primer intent del tractament d'aquest residu. En cas que no sigui la primera vegada modificarà els angles inicials mitjançant la funció `getRandomThetaAndPhi`, per tornar a intentar afegir el residu a la molècula.
- 2- Si és tracta del primer residu de la molècula:
 - a. Si és la primera molècula del sistema:
 - i. Crida a la funció `int2car` de la classe `int2car_m` amb el valor del paràmetre `xyz` buit.
 - ii. Obté el valor dels 3 registres `M` gràcies a la funció `getLast3M`.
 - b. Sinó és la primera molècula del sistema:
 - i. Obté valors aleatori per l'alineació dels angles del primer residu mitjançant la funció `getRandomPositionWithoutCollision`.
- 3- Sinó es tracta del primer residu de la molècula:

- a. Crida a la funció `int2car` de la classe `int2car_m` amb el valor del paràmetre `xyz` igual al valor dels 3 registres `M`.
- b. Obté el valor dels 3 registres `M` gracies a la funció `getLast3M`.

4.1.7. Funció `__addNewResidue`

Paràmetres d'entrada	Descripció
<code>it_atoms</code>	Llista d'àtoms de la molècula actual.
<code>it_xyz</code>	Llista obtinguda amb la funció <code>int2car</code> de la classe <code>int2car_m</code> .
<code>iv_filePREPI</code>	Variable de tipus classe <code>FILEPREPI</code> .
<code>iv_numResidue</code>	Número del residu actual de la molècula.
<code>iv_moleculeID</code>	ID de la molècula actual.
<code>iv_moleculeNumber</code>	Número de la molècula actual.
<code>t_log</code>	Llista amb el resultat del procés.
Paràmetres de sortida	Descripció
<code>lt_atoms</code>	Llista actualitzada dels àtoms.
<code>collision</code>	True si s'ha detectat una col·lisió, False en cas contrari.

Taula 4.1.7.1. Paràmetres d'entrada/sortida

Descripció:

Aquesta funció serveix per afegir nous residus als sistemes `System` i `system2`.

Els passos que fa la funció son:

1. Transforma els valors obtinguts de `int2car` a valors que tenen el format d'un àtom d'un fitxer PDB. Per fer això fa servir dues funcions.
 - a. `convertxyz`
 - b. `convertToAtoms`
2. Comprova si s'han de modificar els àtoms obtinguts a causa de les condicions periòdiques de contorn.
Això ho farà només pel `SYSTEM`, no pel `SYSTEM2`. La comprovació es fa gràcies a la funció `periodicBoundaryCondition`.
3. Comprova si algun d'aquests àtoms del sistema col·lisiona amb alguna molècula de residus anteriors.
Excepció: No fa la comprovació als residus immediatament anteriors.
4. Crea una variable de tipus residu pel `SYSTEM` amb el valor dels àtoms que s'han comprovat les seves condicions periòdiques de contorn.
5. Crea una variable de tipus residu pel `SYSTEM2` amb els àtoms originals, els que no s'ha aplicat les condicions periòdiques de contorn.

6. Esborra les línies DUMM (àtoms “dummys” del fitxer de coordenades) de la llista d'àtoms amb la funció deleteDUMMlines de la classe ATOM.
7. Afegeix el nou residu als sistemes corresponents amb la funció addResidue de la classe MOLECULE.

4.1.8. Funció convertxyz + convertxyz2

Paràmetres d'entrada	Descripció
t_xyz	Llista amb els valors obtinguts amb la funció int2car de la classe INT2CAE_M.
Paràmetres de sortida	Descripció
t_newxyz	Llista amb la conversió dels valors de la llista d'entrada.

Taula 4.1.8.1. Paràmetres d'entrada/sortida de les funcions convertxyz i convertxyz2

Descripció:

La funció “convertxyz” crida a la funció convertxyz2.

L'objectiu d'aquestes 2 funcions és transformar els valors obtinguts amb la funció int2car al format de número amb 8 posicions i 3 decimals.

Dintre de la funció convertxyz va mirant tots els registres de la llista t_xyz i va fent servir la funció convertxyz2 per obtenir els valors convertits a format %8.3f, és a dir 8 posicions amb 3 decimals.

4.1.9. Funció convertToAtoms

Paràmetres d'entrada	Descripció
t_xyz	Llista amb el resultat obtingut amb la funció convertxyz
t_atom_name	Llista amb el nom dels àtoms.
residueName	Nom del residu actual.
residueNumber	Número del residu actual.
chainID	Valor de la cadena.
Paràmetres de sortida	Descripció
t_atoms	Llista amb àtoms.

Taula 4.1.9.1. Paràmetres d'entrada/sortida de la funció convertToAtoms

Descripció:

Els valors obtinguts mitjançant la classe INT2CAR_M s'han transformat amb la funció convertxyz, però ara cal convertir aquesta informació al format que tenen els àtoms dintre de cada línia dels fitxers PDB.

La funció llegeix tota la llista t_xyz i per cada registre genera una variable de tipus àtom i l'afegeix a la llista de sortida t_atoms.

4.1.10. Funció printErrors

Paràmetres d'entrada	Descripció
t_errors	Llista dels errors que s'han detectat.
Lenguaje	Llenguatge en què es mostraran els missatges.

Taula 4.1.10.1. Paràmetres d'entrada de la funció printErrors

Descripció:

Mostrarà per pantalla la llista de missatges que conté la taula t_errors.

4.1.11. Funció getLast3M

Paràmetres d'entrada	Descripció
mLines	Línies amb valor M.
t_xyz	Valor actual XYZ.
numResidue	Número del residu actual.
t_old_last3m	Antic valor last3M.
Paràmetres de sortida	Descripció
t_last3M	Valor actual last3M.

Taula 4.1.11.1. Paràmetres d'entrada de la funció getLast3M

Descripció:

Obté el valor actual del darrers registres que tenen el valor M. On M es tracte dels àtoms de la cadena principal del residu.

4.1.12. Funció PeriodicBoundaryCondition

Paràmetres d'entrada	Descripció
Box	Dimensions de la caixa. box[0]: Dimension X. box[2]: Dimension Y. box[4]: Dimension Z
posX	Posició X a comprovar.
posY	Posició Y a comprovar.
posZ	Posició Z a comprovar.

Paràmetres de sortida	Descripció
posX	Noves posicions (si s'han modificat, sinó seran els mateixos valors que a l'entrada).
posY	
posZ	

Taula 4.1.12.1. Paràmetres d'entrada/sortida de PeriodicBoundaryCondition

Descripció:

Comprova si a les posicions X, Y, Z de l'entrada s'ha d'aplicar les condicions periòdiques de contorn. Això dependrà de les posicions i de les dimensions de la caixa.

Per detectar si cal aplicar el PBC considerem que el punt inicial del sistema serà al mig de la capsa. És a dir a la meitat de totes les seves dimensions. Per tant amb la funció que veiem al codi podrem comprovar si s'ha d'aplicar PBC en algun dels eixos XYZ introduïts com a paràmetres d'entrada.

4.1.13. Funció `__checkCollisionInSystem`

Paràmetres d'entrada	Descripció
moleculeID	ID de la molècula del àtom actual.
atom	Variable de tipus classe ATOM, per tant contindrà tots els atributs de la classe ATOM.
t_log	Llista amb el resultat del procés.
Paràmetres de sortida	Descripció
True o False	True (si ha hagut col·lisió) o False (no hi ha hagut col·lisió).

Taula 4.1.13.1. Paràmetres d'entrada/sortida de la funció `__checkCollisionInSystem`

Descripció:

Comprova si la posició XYZ del àtom col·lisiona amb un altre àtom del sistema.

Per fer això comprova tots els àtoms de tots els residus de totes les molècules que s'estan tractant en el sistema en aquest punt.

Hi ha una excepció. Si s'està comprovant la mateixa molècula (mateix identificador molecule ID) i el mateix residu (mateix residue number) és immediatament anterior a l'actual, no es realitzarà la comprovació de la col·lisió, ja que és evident que col·lisionen, ja que els dos residus estan units entre si.

Una col·lisió es produirà si la distancia entre els àtoms és inferior a la indicada en el paràmetre `minimum_distance` del fitxer.

Per calcular la distancia entre àtoms es fan servir funcions de la classe `numpy`.

```
posicio1 = numpy.array(x1, y1, z1)
posicio2 = numpy.array(x2, y2, z2)
```

```
distancia = numpy.linalg.norm(posicio1-posicio2)
```

Taula 4.1.13.2. Funcions

En cas de trobar alguna col·lisió mostrarà un missatge per pantalla i no continuarà el tractament del sistema.

4.1.14. Funció `__checkCollisionInSystem2`

Paràmetres d'entrada	Descripció
posX	Posició X a validar.
posY	Posició Y a validar.
posZ	Posició Z a validar.
Paràmetres de sortida	Descripció
True o False	True (si ha hagut col·lisió) o False (no hi ha hagut col·lisió).

Taula 4.1.14.1. Paràmetres d'entrada/sortida de la funció `__checkCollisionInSystem2`

Descripció:

Comprova si la posició XYZ del àtom té col·lisió amb un altre àtom del sistema.

Per fer això comprova tots els àtoms de tots els residus de totes les molècules que s'estan tractant en el sistema en aquest punt.

Una col·lisió es produirà si la distancia entre els àtoms és inferior a la indicada en el paràmetre `minimum_distance` del fitxer.

Per calcular la distancia entre àtoms es fan servir funcions de la classe `numpy`.

En cas de trobar alguna col·lisió mostrarà un missatge per pantalla i no continuarà el tractament del sistema.

4.1.15. Funció `__getRandomPositionWithoutCollision`

Paràmetres de sortida	Descripció
x	Posició X on no hi ha col·lisió.
y	Posició Y on no hi ha col·lisió.
z	Posició Z on no hi ha col·lisió.
True o False	True (si ha hagut col·lisió) o False (no hi ha hagut col·lisió).

Taula 4.1.15.1. Paràmetres de sortida de la funció `__getRandomPositionWithoutCollision`

Descripció:

Anirà generant posicions aleatòries fins que passin dues coses:

- Troba una posició que no té col·lisió amb cap àtom.
- La comprovació de col·lisió es realitza amb la funció `checkCollisionInSystem2`.
- Supera el nombre de cops (paràmetre d'entrada "times"), que té per obtenir una posició.

Els nombres aleatoris s'obtenen gràcies a la funció `uniform` de la classe `random`.

4.1.16. Funció `getRandomThetaAndPhi`

Paràmetres d'entrada	Descripció
<code>t_par</code>	Llista amb els valors <code>t_par</code> d'un fitxer PREPI.
<code>maximum_deviation</code>	Desviació màxima.
Paràmetres de sortida	Descripció
<code>t_new_par</code>	Llista amb els valors <code>t_par</code> modificats.

Taula 4.1.16.1. Paràmetres d'entrada/sortida de la funció `getRandomThetaAndPhi`

Descripció:

Per cada registre de la llista `t_par` tenim 3 valors, `r`, `Theta` i `Phi`. Aquesta funció serveix per obtenir valors aleatoris i assignar-los als valors on estan emmagatzemats `Theta` i `Phi`.

Els nombres aleatoris s'obtenen gràcies a la funció `uniform` de la classe `RANDOM`.

4.1.17. Funció `getSystem`

Paràmetres de sortida	Descripció
<code>v_system</code>	Variable de tipus <code>SYSTEM</code> .

Taula 4.1.17.1. Paràmetres d'entrada/sortida de la funció `getSystem`

Descripció:

Retorna una variable de tipus `SYSTEM` per poder tractar els seus atributs.

4.2. INT2CAR_M

A continuació es detallen les funcions de la classe `int2car_M` anomenant els seus paràmetres d'entrada i sortida juntament amb la descripció de la funció i el codi utilitzat

4.2.1. Funció `int2car_x`

Paràmetres d'entrada	Descripció
<code>dist</code>	Distància.
<code>alpha</code>	Angle.

Paràmetres de sortida	Descripció
X	Coordenada x.

Taula 4.2.1.1. Paràmetres d'entrada/sortida de la funció int2car_x

Descripció:

Obté el valor de la posició X.

4.2.2. Funció int2car_y

Paràmetres d'entrada	Descripció
dist	Distància.
alpha	Angle.
dihed	Diedre.
Paràmetres de sortida	Descripció
y	Coordenada y.

Taula 4.2.2.1. Paràmetres d'entrada/sortida de la funció int2car_y

Descripció:

Obté el valor de la posició Y.

4.2.3. Funció int2car_z

Paràmetres d'entrada	Descripció
dist	Distància.
alpha	Angle.
dihed	Diedre.
Paràmetres de sortida	Descripció
Z	Coordenada z.

Taula 4.2.3.1. Paràmetres d'entrada/sortida de la funció int2car_z

Descripció:

Obté el valor de la posició z.

4.2.4. Funció new_car

Paràmetres d'entrada	Descripció
Dist	Distància.
Alpha	Angle.
Dihed	Diedre.
pC	Àtom C.
pB	Àtom B.
pA	Àtom C.
Paràmetres de sortida	Descripció

Newcoord Noves coordenades.

Taula 4.2.4.1. Paràmetres d'entrada/sortida de la funció new_car

Descripció:

Obté el valor de les noves coordenades.

4.2.5. Funció print_xyz

Paràmetres d'entrada	Descripció
title	Títol.
names	Noms.
Coord	Coordenades.

Taula 4.2.5.1. Paràmetres d'entrada de la funció print_xyz

Descripció:

Mostra per pantalla el contingut dels paràmetres d'entrada.

4.2.6. Funció int2car

Paràmetres d'entrada	Descripció
con	Llista amb el contingut de con. con[0]: na. con[1]: nb. con[2]: nc.
par	Llista amb el contingut de par. par[0]: r. par[1]: theta. par[2]: phi.
xyz	Llista amb els valor xyz actuals.
Paràmetres de sortida	Descripció
xyz	Llista amb el valor dels nous xyz.

Taula 4.2.6.1. Paràmetres d'entrada/sortida de la funció int2car

Descripció:

Amb les llistes con, par i xyz actual, obté un nou valor per la llista de sortida xyz.

4.3. FILES

A continuació es detallen les funcions de la classe FILES anomenant els seus paràmetres d'entrada i sortida juntament amb la descripció de la funció.

4.3.1. Funció __init__

Paràmetres d'entrada	Descripció
fileName	Nom del fitxer

Taula 4.3.1.1. Paràmetres d'entrada de la funció __init__

Descripció:

Permet crear una variable de tipus classe files.

Aquesta funció comprova si l'extensió del fitxer és correcta. És a dir, si es tracta d'un fitxer amb extensió .pdb o .prepi. Si no és una d'aquestes extensions aleshores mostrarà el missatge d'error "Error in format file" i no continuarà el procés.

Un cop determinada l'extensió del fitxer procedirà a cridar al mètode __init__ del tipus de fitxer corresponent.

- PDB: Classe filePDB.
- PREPI: Classe filePREPI.

4.3.2. Funció readFile

Paràmetres d'entrada	Descripció
systemName	Nom del fitxer amb la seva extensió. Exemple: ALA.PDB
Paràmetres de sortida	Descripció
v_system	Si es tracta d'un fitxer PDB retorna el sistema que s'ha creat a partir del fitxer PDB.
v_PREPIlines	Si es tracta d'un fitxer PREPI retorna les línies contingudes dintre del fitxer PREPI.

Taula 4.3.2.1. Paràmetres d'entrada de la funció readFile

Descripció:

Permet llegir el contingut del fitxer.

En funció del tipus de fitxer realitza unes accions determinades:

- Fitxer tipus PDB: Crida a la funció readFile de la classe FILEPDB. Aquesta funció permet obtenir el sistema i és el valor de retorn d'aquesta funció.
- Fitxer tipus PREPI: Crida a la funció readFile de la classe FILEPREPI; i obté les línies del fitxer i les retorna per paràmetre.
- Fitxer d'altre tipus: Mostra missatge d'error "Error in format file".

4.3.3. Funció storeFile

Paràmetres d'entrada	Descripció
fileName	Nom del fitxer
System	Sistema

Taula 4.3.3.1. Paràmetres d'entrada de la funció storeFile

Descripció:

Permet emmagatzemar el contingut del fitxer.

Aquesta funció és estàtica “@staticmethod”, per la qual cosa no caldrà fer servir una variable del tipus classe FILES.

En funció del tipus de fitxer realitza unes accions determinades:

- Fitxer tipus PDB: Crida a la funció storeFile de la classe FILEPDB.
- Fitxer tipus PREPI: Crida a la funció storeFile de la classe FILEPREPI.
- Fitxer d'altre tipus: Mostra missatge d'error “Error in format file”.

4.3.4. Funció printFile

Descripció:

Permet mostrar per pantalla el contingut del fitxer.

En funció del tipus de fitxer realitza unes accions determinades:

- Fitxer tipus PDB: Llegeix totes les línies del fitxer que estan emmagatzemades a la variable filePDB.t_file_lines i les imprimeix per pantalla.
- Fitxer tipus PREPI: Llegeix totes les línies del fitxer que estan emmagatzemades a la variable filePREPI.t_file_lines i les imprimeix per pantalla.
- Fitxer d'altre tipus: Mostra missatge d'error “Error in format file”.

4.4. FILEPDB

A continuació es detallen les funcions de la classe FILEPDB anomenant els seus paràmetres d'entrada i sortida juntament amb la descripció de la funció i el codi utilitzat.

4.4.1. Funció readFile

Paràmetres d'entrada	Descripció
fileName	Nom del fitxer.
systemName	Nom del sistema.
Paràmetres de sortida	Descripció
v_system	Variable de tipus classe SYSTEM.

Taula 4.4.1.1. Paràmetres d'entrada de la funció readFile

Descripció:

Funció per llegir els fitxer de tipus PDB.

Realitza aquests punts:

1. Primer comprova que el fitxer realment existeixi, en cas contrari mostrarà per pantalla el missatge d'error "Could not read file: #nom del fitxer#" i no continuarà el procés.
2. Llegeix el contingut del fitxer i guarda en la variable t_file_lines només les línies que contenen el text ATOM en les 6 primeres posicions de la línia.
3. Si no ha trobat cap línia amb el text ATOM, mostrarà el missatge d'error "No PDB lines found in file" i no continuarà el procés.
4. Crearà la variable v_system de tipus classe SYSTEM. Li assignarà el paràmetre d'entrada systemName com a nom del sistema.
5. A continuació llegirà totes les línies i realitzarà el següent:
 - a. Si és una nova molècula afegirà al sistema una nova variable de tipus classe MOLECULE.
 - b. Si és un nou residu afegirà un nou residu a la molècula amb la funció addResidue de la classe MOLECULE.
 - c. Crearà una nova variable de tipus classe ATOM per cada nova línia i l'afegirà a la llista t_atom.

4.4.2. Funció storeFile

Paràmetres d'entrada	Descripció
fileName	Nom del fitxer.
System	Variable de tipus classe SYSTEM.

Taula 4.4.2.1. Paràmetres d'entrada de la funció storeFile

Descripció:

Permet guardar en format fitxer el contingut de la classe FILEPDB.

Realitza aquests punts:

1. Primer comprova si existeix algun fitxer amb el mateix nom, si es dona el cas mostrarà el missatge d'error "File #nom del fitxer# exists" i no continuarà el procés.
2. Per totes les molècules del sistema realitzarà el següent:
 - a. Llegirà tots els residus i per cada residu farà:
 - i. Llegirà tots els seus àtoms i per cada àtom crearà una nova línia del fitxer gracies a la funció addLine de la classe ATOM.
 - ii. Si es tracta del darrer àtom del sistema crearà una línia final de fitxer gracies a la funció addEndLine de la classe ATOM.

4.4.3. Funció addLine

Paràmetres d'entrada	Descripció
atom	Variable de tipus classe ATOM.
serialNumber	Número de sèrie.
Paràmetres de sortida	Descripció
line	Línia amb tota la informació de l'àtom.

Taula 4.4.3.1. Paràmetres d'entrada de la funció addLine

Descripció:

Aquesta funció obté les dades dels atributs de ATOM, afegeix el valor del serialNumber i retorna tota la informació en una única variable.

Aquesta variable servirà per crear una nova línia al fitxer PDB.

El resultat serà similar a:

```
ATOM      20  O      ALA  A      2           7.425   6.222   0.000   1.00   0.00      O
```

4.4.4. Funció addEndLine

Paràmetres d'entrada	Descripció
atom	Variable de tipus classe ATOM.
serialNumber	Número de sèrie.
Paràmetres de sortida	Descripció
line	Línia amb la informació de l'àtom necessària per indicar que es una línia de fi de molècula.

Taula 4.4.4.1. Paràmetres d'entrada de la funció addEndLine

Descripció:

Aquesta funció es similar a l'anterior, addLine, però aquesta es fa servir per crear una línia de tipus fi de molècula. Es farà servir en el moment de la creació del fitxer PDB.

Llegirà el paràmetre d'entrada serialNumber, els atributs residue_name, chain_id i residue_number i generarà una línia similar a:

```
TER      151      ALA A   15
```

4.5. FILEPREPI

A continuació es detallen les funcions de la classe FILEPREPI anomenant els seus paràmetres d'entrada i sortida juntament amb la descripció de la funció i el codi utilitzat

4.5.1. Funció __init__

Paràmetres d'entrada	Descripció
Self	Nom del fitxer

Taula 4.5.1.1. Paràmetres d'entrada de la funció __init__

Descripció:

Permet crear una variable de tipus classe FILEPREPI.

4.5.2. Funció readFile

Paràmetres d'entrada	Descripció
fileName	Nom del fitxer

Taula 4.5.2.1. Paràmetres d'entrada de la funció readFile

Descripció:

Funció per llegir els fitxer de tipus PREPI.

Realitza aquests punts:

1. Primer comprova que el fitxer realment existeixi, en cas contrari mostrarà per pantalla el missatge d'error "Could not read file: #nom del fitxer#" i no continuarà el procés.
2. Emmagatzema el contingut del fitxer en la llista de línies del fitxer.
3. Crida a la funció trataFile que realitza la resta del procés.
4. Comprova si existeix el fitxer

4.5.3. Funció trataFile

Llegeix les línies emmagatzemades a `t_file_lines` i:

1. Omple la llista `t_name` amb el contingut del nom de cada línia.
2. Realitza crides a les funcions `getCar` i `getPar`.
3. Omple la llista `t_mLines` amb les línies del fitxer PREPI que son tipus M.

4.5.4. Funció getCar

Paràmetres d'entrada	Descripció
na	Valor na de la línia del fitxer PREPI.
nb	Valor nb de la línia del fitxer PREPI.
nc	Valor nc de la línia del fitxer PREPI.
Paràmetres de sortida	Descripció
t_aux	Llista amb els valors na, nb, nc

Taula 4.5.4.1. Paràmetres d'entrada de la funció trataFile

Descripció:

Llegeix el valor dels paràmetres na, nb i nc i crea la llista `t_aux` amb aquests valors.

4.5.5. Funció getPar

Paràmetres d'entrada	Descripció
R	Distància. És el valor <i>r</i> de la línia del fitxer PREPI.
Theta	Angle. És el valor theta de la línia del fitxer PREPI.
Phi	Diedre. És el valor phi de la línia del fitxer PREPI.
Paràmetres de sortida	Descripció
t_aux	Llista amb els valors r, theta i phi

Taula 4.5.5.1. Paràmetres d'entrada de la funció getPar

Descripció:

Llegeix el valor dels paràmetres r, theta i phi, i crea la llista `t_aux` amb aquests valors.

4.5.6. Funció checkFile

Descripció:

Mostra per pantalla el contingut dels atributs de la classe:

- Residue name.
- Atoms name.

- Con.
- Par.
- M lines.

4.5.7. Funció storeFile

Descripció:

Aquesta funció ara mateix no fa res, però si en un futur es vol implementar es podrà fer. S'ha creat perquè a la classe FILEPDB si que funciona storeFile i des de la classe FILES es crida a una classe o l'altra en funció de la classe.

4.6. LINEPREPI

A continuació es detallen les funcions de la classe LINEPREPI anomenant els seus paràmetres d'entrada i sortida juntament amb la descripció de la funció i el codi utilitzat

4.6.1. Funció __init__

Paràmetres d'entrada	Descripció
line	Línia d'un fitxer PREPI.

Taula 4.6.1.1. Paràmetres d'entrada de la funció __init__

Descripció:

Aquesta funció la fa servir la classe FILEPREPI per obtenir d'una línia del fitxer PREPI, el valor emmagatzemat a les seves columnes: Number, igrph, isymb1, tree, na, nb, nc, r, theta, phi i chrg.

4.7. MESSAGES

A continuació es detallen les funcions de la classe MESSAGES anomenant els seus paràmetres d'entrada i sortida juntament amb la descripció de la funció i el codi utilitzat.

4.7.1. Funció getTextMessage

Paràmetres d'entrada	Descripció
numMessage	Número del missatge.
language	Idioma del missatge d'error. Els idiomes s'identifiquen per dos lletres. Per exemple EN = Anglès.

Paràmetres de sortida	Descripció
message	Valor associat al missatge

Taula 4.7.1.1. Paràmetres d'entrada de la funció `getTextMessage`

Descripció:

Retornarà el text del missatge associat al numero de missatge i a l'idioma sol·licitat.

4.8. SYSTEM

A continuació es detallen les funcions de la classe SYSTEM anomenant els seus paràmetres d'entrada i sortida juntament amb la descripció de la funció i el codi utilitzat

4.8.1. Funció `__init__`

Paràmetres d'entrada	Descripció
name	Nom del sistema.
minimum_distance	Distància mínima a la qual es considerarà que s'ha produït una col·lisió.
maximum_deviation	Desviació mínima.
max_retries	Nombre d'intents màxims que s'intentarà situar un residu per evitar la col·lisió.
box	Dimensions i unitats de la caixa on esta la molècula.
language	Llenguatge amb què es mostraran els missatges.

Taula 4.8.1.1. Paràmetres d'entrada de la funció `__init__`

Descripció:

Permet crear una variable de tipus classe SYSTEM.

4.8.2. Funció `addMolecule`

Paràmetres d'entrada	Descripció
molecule	Molècula.

Taula 4.8.2.1. Paràmetres d'entrada de la funció `addMolecule`

Descripció:

Permet afegir una nova molècula a la llista `t_molecules`, que es un dels atributs de la classe. El paràmetre d'entrada ha de ser del tipus classe MOLECULE.

4.8.3. Funció printSystemNumberElements

Descripció:

Aquesta funció serveix per mostrar per pantalla dades del sistema, per totes les molècules del sistema mostrà per pantalla:

- Molecule ID.
- Numero de residus.
- Numero d'àtoms.

4.8.4. Funció printSystemMolecules

Descripció:

Aquesta funció serveix per mostrar per pantalla dades del sistema.

- Nom del sistema.
- Per cada molècula del sistema farà servir la funció printMolecule de la classe MOLECULE.

4.9. MOLECULE

A continuació es detallen les funcions de la classe MOLECULE anomenant els seus paràmetres d'entrada i sortida juntament amb la descripció de la funció i el codi utilitzat

4.9.1. Funció __init__

Paràmetres d'entrada	Descripció
t_residues	Llista amb els residus que té la molècula.
name	Nom de la molècula.
ID	ID de la molècula.

Taula 4.9.1.1. Paràmetres d'entrada de la funció __init__

Descripció:

Permet crear una variable de tipus classe MOLECULE.

Indicarem el nom de la molècula, el seu ID i també la llista de residus que tindrà aquesta molècula.

Si posteriorment es volen afegir mes residus a la molècula es pot fer servir la funció addResidue.

4.9.2. Funció addResidue

Paràmetres d'entrada	Descripció
residue	Valors d'un residu. El tipus de la variable ha de ser de classe RESIDUE.

Taula 4.9.2.1. Paràmetres d'entrada de la funció addResidue

Descripció:

Permet afegir un nou residu a la molècula. Aquest residu s'afegirà a la llista de residus que estan dintre de l'atribut de la classe t_residues.

4.9.3. Funció printMolecule

Descripció:

Mostrarà per pantalla les dades que formen la molècula, per fer això anirà llegint tots els residus emmagatzemats a la llista t_residues i anirà cridant a la funció printResidue que es troba a la classe RESIDUE.

4.9.4. Funció getNextChainID

Paràmetres d'entrada	Descripció
actualChainID	Valor actual del ChainID de la molècula.
Paràmetres de sortida	Descripció
nextChainID	Valor següent del valor actual de la molècula.

Taula 4.9.4.1. Paràmetres d'entrada de la funció getNextChainID

Descripció:

El Chain ID es el ID de la molècula i es una lletra d'abecedari que va incrementant el seu valor. Comença per A, després B, etc.

Aquesta funció serveix per obtenir el següent valor a partir d'un valor inicial.

Si aquest valor inicial està en blanc o es tracta d'una Z, obtindrà el valor A, sinó obtindrà el següent valor del abecedari.

4.10. RESIDUE

A continuació es detallen les funcions de la classe RESIDUE anomenant els seus paràmetres d'entrada i sortida juntament amb la descripció de la funció i el codi utilitzat

4.10.1. Funció `__init__`

Paràmetres d'entrada	Descripció
<code>t_atom</code>	Llista amb els àtoms que formen el residu.
<code>name</code>	Nom del residu.
<code>number</code>	Numero del residu.
<code>moleculeID</code>	ID de la molècula a la que pertany el residu.

Taula 4.10.1.1. Paràmetres d'entrada de la funció `__init__`

Descripció:

Permet crear una variable de tipus classe RESIDUE. Amb aquesta variable podrem fer servir la resta de funcions de la classe.

Li podrem indicar els àtoms inicials, el nom, el número i a quin ID de molècula pertany.

4.10.2. Funció `printResidue`

Descripció:

Aquesta funció mostrarà per pantalla les dades del residu.

4.10.3. Funció `addAtom`

Paràmetres d'entrada	Descripció
<code>atom</code>	Variable tipus classe ATOM.

Taula 4.10.3.1. Paràmetres d'entrada de la funció `addAtom`

Descripció:

Afegirà un nou àtom a la llista `t_atom` que forma part dels atributs de la classe.

4.10.4. Funció `getAtom`

Paràmetres d'entrada	Descripció
<code>atom_name</code>	Nom de l'àtom a cercar.

Taula 4.10.4.1. Paràmetres d'entrada de la funció `getAtom`

Descripció:

Comprovarà si el nom del àtom esta dintre de la llista `t_atom`, que forma par dels atributs de la classe.

Si troba l'àtom, el retornarà en una variable tipus classe `ATOM`, sinó el troba mostrarà el missatge d'error: "Atom nom del àtom Not found".

4.11. ATOM

A continuació es detallen les funcions de la classe `ATOM` anomenant els seus paràmetres d'entrada i sortida juntament amb la descripció de la funció i el codi utilitzat.

4.11.1. Funció `__init__`

Paràmetres d'entrada	Descripció
<code>tipus</code>	Tipus.
<code>serial_number</code>	Número de sèrie.
<code>name</code>	Nom.
<code>chain_id</code>	ID de la cadena.
<code>x</code>	Coordenada X.
<code>y</code>	Coordenada Y.
<code>z</code>	Coordenada Z.
<code>occupancy</code>	Ocupació.
<code>temperature</code>	Temperatura.
<code>element_symbol</code>	Símbol de l'element.
<code>charge</code>	Carga.
<code>residue_name</code>	Nom del residu.
<code>residue_number</code>	Número de residu.

Taula 4.11.1.1. Paràmetres d'entrada de la funció `__init__`

Descripció:

Permet crear una variable de tipus classe `ATOM`. Per això cal indicar-li els valors que tindrà l'àtom.

Aquests valors s'emmagatzemaran com atributs de la classe.

4.11.2. Funció `printAtom`

Descripció:

Mostrarà per pantalla les dades dels atributs de la classe `ATOM`.

Serial number, chain ID, posició X, posició Y, posició Z, occupancy, temperature, element symbol, charge.

4.11.3. Funció deleteDUMMLines

Paràmetres d'entrada	Descripció
t_atom	Taula amb la llista dels àtoms.
Paràmetres de sortida	Descripció
t_atom_without_dumm_lines	La mateixa taula sense línies DUMM

Taula 4.11.3.1. Paràmetres d'entrada de la funció deleteDUMMLines

Descripció:

Llegeix la llista del paràmetre d'entrada t_atom, elimina les línies que son de tipus DUMM i retorna la llista sense aquestes línies.

Les línies tipus DUMM son las que tenen el valor DUMM dintre de l'atribut name que es troba al fitxer PREPI.

5. Eines utilitzades

5.1. Python

Python ^[23] és un llenguatge de programació amb codi obert, molt potent, amigable i fàcil d'aprendre que es pot executar en qualsevol entorn, funciona de forma interactiva i per tant és un programa amb un propòsit general molt utilitzat que pertany al grup d'idiomes com Java, Perl, i PHP.

Els tècnics informàtics que l'utilitzen poden obtenir resultats nous molt ràpidament i es tracta també d'una eina molt utilitzada per a científics que treballen interactivament amb dades.

A més, ha estat molt important que, des del començament, Python hagi estat disponible segons la filosofia de codi obert, fins i tot abans que s'inventés el terme "open code". Tothom pot utilitzar la font lliurement, sempre que els drets d'autor estiguin indicats correctament; i per aquest motiu, els usuaris de Python se senten part d'una comunitat i són molt actius a l'hora de fer millores o crear fòrums en les xarxes on s'explica com millorar els desenvolupaments.

A més de les aplicacions de serveis web també s'utilitza per a la creació ràpida de prototips: ja que a causa de la gran quantitat de biblioteques estàndard amb codi reutilitzable es poden crear ràpidament noves aplicacions. Per exemple s'utilitza entre altres coses en el desenvolupament web (per exemple. Zope), jocs (Star Trek Pont Commander), gràfics ("L'amença fantasma" i "El retorn de la mòmia"), bancs, ciència, educació, desenvolupament de programari (Red Hat) i programari empresarial (RealNetworks).

5.1.1. Història del Python

Va ser creat per l'holandès "Guido Van Rossum" el desembre del 1989 en els Països Baixos dins el "Centrum Wiskunde & Informatica"(CWI o Institut de recerca nacional per ciències matemàtiques i informàtica als països baixos) i va acabar el primer esborrany de treball uns mesos després en el 1990 (abans del primer llançament públic de la versió 0.9.0 del 20 de febrer de 1991), tot i que la idea ja va ser concebuda a finals de la dècada del 1980.

Es va crear com a successor del llenguatge de programació "ABC" (també desenvolupat en el CWI i creat com una alternativa al llenguatge BASIS).

Van Rossum és el principal autor de Python, i el seu permanent rol central en les decisions és reflecteix en el sobrenom que li ha posat la comunitat Python: "Benevolent dictator for life (BDFL)"

5.1.2. Programació amb Python

Es podria dir que es tracta d'un llenguatge de programació d'alt nivell que s'implementa d'una manera que posa èmfasi en la interactivitat. Python comparteix algunes característiques amb llenguatges de script, però també comparteix algunes característiques amb llenguatges de programació més tradicionals.

Com s'ha comentat és un llenguatge de codi obert "multiparadigma", ja que suporta orientació a objectes, programació imperativa, programació funcional i es basa en scripts (interpretat). Això vol dir, que el codi s'executa en intèrpret en comptes de ser compilat, cosa que estalvia temps en el desenvolupament del programa. D'altra banda en disposar d'un mode interactiu, permet executar el codi en temps real segons es van escrivint les línies de codi per poder visualitzar els resultats.

També es tracta d'un llenguatge de programació multiplataforma, que es pot utilitzar en diferents plataformes com ara Microsoft Windows, GNU/Linux, Unix, MacOS, Solaris

No és necessari declarar el tipus de dada que contindrà una determinada variable, sinó que aquesta dada serà determinada en el moment o temps d'execució segons el valor que se li assigni a la variable. D'altra banda cal tenir en compte que si una variable és de tipus enter, no es podrà tractar com a cadena.

El Python està dissenyat per ser un llenguatge molt visual, i, com a característica principal i distintiva, utilitza sagnat. Això és poc comú en llenguatges de programació, ja que molts (com el C o Java) utilitzen delimitadors com poden ser els caràcters "{" i "}". I una altra característica és el sagnat, ja que és obligatori i s'utilitza per delimitar blocs. És a dir, si augmenta el sagnat precedit d'una declaració correcta, significa l'inici d'un nou bloc; i al contrari, la disminució significa el final del bloc.

5.1.3. Llibreries de Python

Per poder realitzar el desenvolupament del programa han sigut necessari incorporar un conjunt de llibreries específiques ja existents en Python, que vénen incloses per defecte en la versió estàndard. Amb l'ajuda d'aquestes biblioteques s'ha facilitat l'elaboració del programa podent cridar a vàries funcions desenvolupades i d'accés lliure i gratuïtes.

El projecte s'ha desenvolupat utilitzant la versió 2.7 de Python tot i que actualment ja existeix la versió 3.6. que en principi ha de ser el futur del llenguatge en Python.

S'ha utilitzat la versió 2.X que no és compatible amb les 3.X pel fet que té més llibreries de suport que la nova versió; i que han estat utilitzades per simplificar el codi del programa.

A continuació s'enumeren les biblioteques utilitzades amb una breu descripció de les mateixes i la seva utilització dins del programa.

5.1.4. Sci.py

Scipy és una llibreria amb contingut científic i d'enginyeria per a Python distribuïda de forma lliure. Permet utilitzar funcions pròpies del camp de la ciència i l'enginyeria com ara poden ser: funcions de càlcul numèric, interpolació, integració, resolució d'equacions diferencials ordinàries, estadística descriptiva, àlgebra lineal i altres molts aspectes^[30].

5.1.5. Num.py

NumPy és un mòdul contingut dins de Scipy; i es tracta de la llibreria científica principal per al Python sota la llicència BSD, el que permet poder reutilitzar totes les seves funcions^[31].

És una eina de càlcul i creació de vectors multidimensionals i conté llibreries matemàtiques d'alt nivell que permeten realitzar operacions amb els vectors. Per exemple es poden enumerar alguns dels seus continguts:

- Eines per a la integració del codi C / C++ i Fortran
- Funcions sofisticades (de radiodifusió)
- Objectes de matriu N-dimensional
- Funcions d'àlgebra lineal útil, transformació de Fourier i capacitats de nombres aleatoris

5.1.6. Sys.py

La llibreria sys proporciona informació sobre constants, funcions i mètodes del Python interpreter. Per exemple `dir(System)` proporciona un resum de les constants, funcions i mètodes disponibles; o una altra possibilitat és la funció `help()`, que evidentment proporciona informació detallada^[32].

En el nostre cas s'utilitza per sortir de la funció sense continuar "sys.exit"

5.1.7. Pdb.py

La llibreria Pdb és un "debugger" de codi font interactiu per als programes de Python

Inclou funcions que us permeten aturar el programa que s'està escrivint, consultar els valors de les variables i veure l'execució del programa pas a pas, de manera que es pot comprendre el que realment fa i trobar errors a la lògica^[33].

5.1.8. Os.py

Aquesta llibreria proporciona una forma d'utilitzar el sistema operatiu dependent funcions successivament. Per exemple ha estat utilitzada per comprovar que el path del fitxer existeix^[34].

A continuació es llisten algunes de les funcions de la llibreria:

- Execució d'un comandament de shell `os.system ()`
- Obtenir l'entorn dels usuaris `os.environ ()` #
- Torna el directori de treball actual. `os.getcwd ()`
- Retorna l'id del grup real del procés actual. `os.getgid ()`
- Retorna l'identificador d'usuari del procés actual. `os.getuid ()`
- Retorna l'ID del procés real del procés actual. `os.getpid ()`
- Estableix l'umask numèric actual i torna el umask anterior. `os.umask (màscara)`
- Retorna informació que identifica el sistema operatiu actual. `os.uname ()`
- Canviar el directori arrel del procés actual a la ruta d'accés. `os.chroot`
- Retorna una llista de les entrades que es donen al directori per la ruta d'accés. `os.listdir`
- Crea un directori anomenat ruta amb el mode numèric. `os.mkdir`
- Funció recursiva de creació de directori. `os.makedirs`
- Suprimeix la ruta del fitxer. `os.remove`
- Elimina els directoris de forma recursiva. `os.removedirs (ruta d'accés)`
- Canviar el nom del fitxer o el directori `src` a `dst`. `os.rename (src, dst)`
- Suprimeix la ruta del directori. `os.rmdir (ruta d'accés)`

5.2. Py-Charm Edu

PyCharm és un entorn integrat de desenvolupament (IDE) utilitzat en la programació d'ordinadors, específicament per al llenguatge Python. Està desenvolupat per l'empresa txeca JetBrains i és de lliure accés^[35].

Proporciona l'anàlisi de codi, un depurador gràfic, un verificador d'unitat integrat, la integració amb sistemes de control de versions (VCS) i suporta el desenvolupament web amb Django.

S'ha utilitzat per a escriure el codi del programa, ja que es tracta d'una eina que presenta molts avantatges:

- PyCharm és multiplataforma, amb versions Windows, MacOS i Linux.
- Possibilitat de codificació i anàlisi, comprovació de sintaxi i ressaltat d'errors, integració i correccions ràpides.

- Navegació per codi: visites de projectes especialitzats, visites d'estructura de fitxers i salt ràpid entre fitxers, classes, mètodes i usos
- Ajuda en la programació amb Python: inclou el nom, el mètode d'extracció, la introducció de la variable, la introducció constant, el pull up, el push down i altres
- Proves d'unitat integrades, amb cobertura de codi lineal

5.3. Jmol

Es tracta d'un visor tridimensional de models moleculars gratuït que permet realitzar manipulacions i càlculs a partir d'aquests, i que s'ha utilitzat per comprovar la coherència dels fitxers PDB de sortida del programa^[36].

Està escrit en llenguatge Java de codi obert per a estructures químiques en tres dimensions amb una llicència GNU/PL sotmès a constants revisions i actualitzacions del programari.

Els motius per al que s'ha utilitzat aquest programa i no altres visors són els següents:

- Es pot descarregar de forma gratuïta d'internet molt ràpidament; i es tracta d'un programa autònom en Java que funciona localment en l'ordinador, fora de xarxa.
- Es compatible amb tots els sistemes operatius (Windows, Mac OS X y Linux/Unix) i amb tots els navegadors.
- No hi ha requeriments previs a la seva instal·lació i en cas de necessitat es podria executar sense aquesta des de la memòria d'un USB.
- Està disponible en diverses llengües, com ara el català, i existeixen nombrosos manuals gratuïts
- Té un funcionament molt intuïtiu i és molt configurable, però al mateix temps es poden realitzar mesures d'angles i distàncies d'enllaç químic.
- Es compatible amb la major part dels formats utilitzats en química per la codificació de molècules com ara els fitxers de sortida que s'utilitzen en aquest projecte: PDB.

5.4. Git

Git és un sistema de control de versions distribuït lliure i de codi obert dissenyat per gestionar tot, des de projectes petits fins a grans, amb rapidesa i eficiència^[37].

En principi és fàcil d'aprendre i té una petita corba d'aprenentatge si es compara amb el rendiment obtingut.

Va ser dissenyat per Linus Torvalds (dissenyador de Linux), pensat en l'eficiència i confiança de manteniment de versions d'aplicacions amb una enorme quantitat de fitxers de codi font.

S'ha utilitzat en projecte per anar emmagatzemant les diferents versions del codi Python realitzades per tots els participants del projecte, permet treballar a tothom a la vegada sense cometre errors en les versions desenvolupades.

Els motius per al que s'ha utilitzat aquest programa i no altres visors són els següents:

- Codi lliure i obert: Publicat sota llicència GNU per garantir la llibertat de compartir i canviar programari lliure.
- Permet realitzar diverses branques de desenvolupament gestionat amb canvis de versions.
- És molt ràpid, ja que la majoria de les operacions es realitzen localment. La velocitat i el rendiment ha estat sempre l'objectiu de l'aplicació.
- Es pot realitzar una clonació del repositori complet, així com múltiples còpies de seguretat per a cada usuari que l'utilitzi o qualsevol flux de treball.
- Seguretat: El model de dades que utilitza Git assegura la integritat criptogràfica de cada bit del projecte.
- Disposa d'una àrea de "índex" on es poden formatar o revisar les actualitzacions abans de realitzar l'actualització del repositori de software remot.
- Disponibilitat de manuals i documentació gratuïta a la xarxa.

5.5. Tortoise

TortoiseGit és una interfície de Windows Shell a Git i basada en TortoiseSVN que s'ha utilitzat per realitzar les actualitzacions dels desenvolupaments de Python^[38].

S'ha desenvolupat sota la GPL. El que significa que és totalment gratuït per a qualsevol persona, inclòs en un entorn comercial, sense cap restricció. El codi font també està disponible, de manera que fins i tot es podria desenvolupar una pròpia versió del programa.

Els motius per al que s'ha utilitzat aquest programa i no altres visors són els següents:

- Codi lliure i obert, i per tant gratuït
- Continues actualitzacions amb millores i suport al programari que fa que sigui una eina estable.
- Fàcil d'utilitzar, ja que totes les ordres es troben disponibles des de l'explorador de Windows i sols es mostren aquelles que es poden realitzar en cada una de les situacions.
- Es pot utilitzar amb qualsevol tipus de fitxer.

- Visualització de diferents logs on es pot seguir l'històric de modificacions realitzades en els arxius.
- Disponibilitat en diversos idiomes, això com de manuals i documentació gratuïta a la xarxa.

6. Aplicació

6.1. Mode d'ús

El programa desenvolupat realitza la lectura d'un fitxer de text amb un format determinat on s'especifiquen les característiques del sistema que es vol crear (fitxer input), les ubicacions dels fitxers de lectura amb informació estructural de tots els monòmers que formen el sistema (fitxers prepi), i el fitxer de sortida.

Un cop interpretades les dades i realitzada la lectura dels fitxers estructurals d'entrada, s'inicia el procés de construcció del sistema amb les seves validacions, per finalment si tot és correcte donar com a resultat el sistema desitjat en format PDB.

Per revisar el detall del programa cal adreçar-se al manual adjuntant en l'apartat d'annexos d'aquest mateix document.

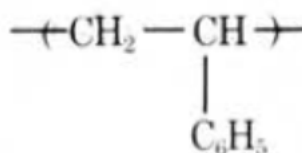
6.2. Exemples d'aplicació

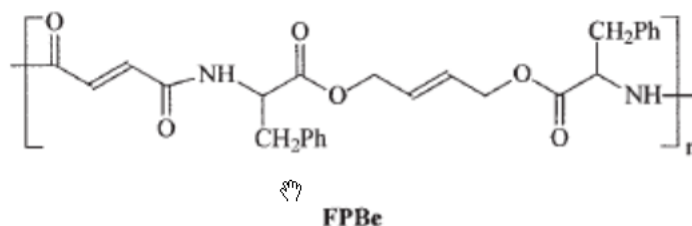
Amb el programa finalitzat i llest per la seva distribució, s'han realitzat varies proves de creació de matrius polimèriques a partir de diferents residus o monòmers. Per corroborar la seva fiabilitat dels resultats oferts per l'aplicació, resta fer una comprovació final on es realitzaran i documentaran algunes proves de generació de polímers.

Es començarà documentant les proves d'un polímer creat a partir de monòmers molt simples com ara el polietilè (PE) i posteriorment es realitzaran altres proves amb polímers creats a partir del poliestirè (PES) i una poliesteramida. Les estructures químiques de cada un dels monòmers utilitzats es poden veure en les següents imatges:



Imatge 6.2.1. Monòmer etilè^[39]



Imatge 6.2.2. Poliestirè^[39]Imatge 6.2.3. Poliesteramida^[40]

6.2.1. Preparació de les dades d'entrada

Les dades d'entrada necessàries per l'execució del programa són:

- Un fitxer de càrrega amb les característiques del sistema a generar.
- Disposar d'un directori amb els fitxers estructurals de cada monòmer del sistema en format PREPI.

A continuació es mostren els tres fitxers d'exemple que s'utilitzaran per fer les càrregues; així com el contingut dels fitxers amb format PREPI dels monòmers necessaris.

6.2.1.1. Fitxers de càrrega

A continuació es mostren imatges dels fitxers de càrrega utilitzats per crear els sistemes. Els fitxers reals es troben en el CD entregat en la memòria estructurats tal i com s'explica en l'annex 4.

En els sistemes de monòmers s'afegeixen varies molècules amb "n" monòmers cada una fins que al afegir més molècules el resultat de l'execució del programa sempre dona col·lisió.

Es mostren part dels fitxers de càrrega utilitzats, el contingut dels fitxers al complet es podran visualitzar en el Annex4.

```
system_name,185PEH_23PE_PETNV3
box,Lx 50 Å,Ly 50 Å,Lz 100 Å
periodical_coordinates,N
minimum_distance, 0.5 Å
retries, 15
maximum_deviation, 5
molecules_number,185
path_prepi,prepis
PEH,23 PE,PET
PEH,23 PE,PET
```



```
...
PEH,23 PE,PET
PEH,23 PE,PET
PEH,23 PE,PET
```

Taula 6.2.1.1.1 Fitxer de càrrega PEH 23PE PET sense coordenades periòdiques

```
system_name,185PEH_23PE_PETNV3
box,Lx 50 A,Ly 50 A,Lz 100 A
periodical_coordinates,Y
minimum_distance, 0.5 A
retries, 15
maximum_deviation, 5
molecules_number,185
path_prepi,prepis
PEH,23 PE,PET
PEH,23 PE,PET
...
PEH,23 PE,PET
PEH,23 PE,PET
PEH,23 PE,PET
```

Taula 6.2.1.1.2 Fitxer de càrrega PEH 23PE PET amb coordenades periòdiques

```
system_name,140PEH22PSPETN
box,Lx 50 A,Ly 50 A,Lz 100 A
periodical_coordinates,N
minimum_distance, 0.5 A
retries, 15
maximum_deviation, 5
molecules_number,185
path_prepi,prepis
PEH,22 PS,PET
PEH,22 PS,PET
...
PEH,22 PS,PET
PEH,22 PS,PET
PEH,22 PS,PET
```

Taula 6.2.1.1.2 Fitxer de càrrega PEH 22PS PET sense coordenades periòdiques

```
system_name,140PEH22PSPETN
box,Lx 50 A,Ly 50 A,Lz 100 A
periodical_coordinates,Y
minimum_distance, 0.5 A
retries, 15
maximum_deviation, 5
molecules_number,185
path_prepi,prepis
PEH,22 PS,PET
PEH,22 PS,PET
...
PEH,22 PS,PET
PEH,22 PS,PET
PEH,22 PS,PET
```

Taula 6.2.1.1.4 Fitxer de càrrega PEH 22PS PET amb coordenades periòdiques



```

system_name,FPB05N
box,Lx 100 A,Ly 100 A,Lz 100 A
periodical_coordinates,N
minimum_distance, 0.5 A
retries, 15
maximum_deviation, 5
molecules_number,70
path_prepi,prepis
HED, 4 FPB, TAI
HED, 4 FPB, TAI
HED, 4 FPB, TAI
HED, 4 FPB, TAI
...
HED, 4 FPB, TAI

```

Taula 6.2.1.1.5 Fitxer de càrrega Poliesteramida sense coordenades periòdiques

```

system_name,FPB05Y
box,Lx 100 A,Ly 100 A,Lz 100 A
periodical_coordinates,Y
minimum_distance, 0.5 A
retries, 15
maximum_deviation, 5
molecules_number,70
path_prepi,prepis
HED, 4 FPB, TAI
HED, 4 FPB, TAI
HED, 4 FPB, TAI
HED, 4 FPB, TAI
...
HED, 4 FPB, TAI

```

Taula 6.2.1.1.6 Fitxer de càrrega Poliesteramida amb coordenades periòdiques

6.2.1.2. Monòmers en format PREPI

Es mostrarà l'aparença dels tres fitxers en format PREPI amb la composició dels monòmers utilitzats per crear les matrius en coordenades esfèriques:

```

PE

MOL    INT    0
CORRECT      OMIT DU    BEG
0.0000
  1  DUMM  DU    M    0  -1  -2    0.000    .0    .0
.00000
  2  DUMM  DU    M    1    0  -1    1.529    .0    .0
.00000
  3  DUMM  DU    M    2    1    0    1.529  109.461    .0
.00000
  4  C3    c3    M    3    2    1    1.529  109.465  178.348 -
0.041182

```

5	H14	hc	E	4	3	2	1.088	108.887	62.420
0.007304									
6	H15	hc	E	4	3	2	1.089	108.382	-62.420
0.020252									
7	C4	c3	M	4	3	2	1.529	109.460	178.348 -
0.023226									
8	H16	hc	E	7	4	3	1.088	108.887	62.420
0.027114									
9	H17	hc	E	7	4	3	1.089	108.382	-62.420
0.009739									
LOOP									
IMPROPER									
DONE									

Taula 6.2.1.2.1 Composició fitxer en format PREPI del monòmer PE

0	0	2										
This is a remark line												
molecule.res												
MOL	INT	0										
CORRECT	OMIT	DU	BEG									
0.0000												
1	DUMM	DU	M	0	-1	-2	0.000	.0	.0			
.00000												
2	DUMM	DU	M	1	0	-1	1.529	.0	.0			
.00000												
3	DUMM	DU	M	2	1	0	1.529	109.461	.0			
.00000												
4	C7	c3	M	3	2	1	1.529	109.461	178.348			
0.061594												
5	H22	hc	E	4	3	2	1.088	109.035	62.420	-		
0.015871												
6	H23	hc	E	4	3	2	1.089	109.317	-62.420	-		
0.008472												
7	C8	c3	M	4	3	2	1.529	109.461	178.348	-		
0.073243												
8	H24	hc	E	7	4	3	1.086	109.035	62.429			
0.008627												
9	H25	hc	E	7	4	3	1.085	109.317	-62.429			
0.010575												
10	H26	hc	E	7	9	8	1.086	109.461	-120.000			
0.016790												
LOOP												
IMPROPER												
DONE												
STOP												

Taula 6.2.1.2.2 Composició fitxer en format PREPI del monòmer PET

0	0	2									
CORRECT		OMIT DU	BEG								
0.0000											
1	DUMM	DU	M	0	-1	-2	0.000	.0	.0		
.00000											
2	DUMM	DU	M	1	0	-1	1.529	.0	.0		
.00000											
3	DUMM	DU	M	2	1	0	1.529	109.461	.0		
.00000											
4	C1	c3	M	3	2	1	1.529	109.461	178.348	-	
0.193871											
5	H9	hc	E	4	3	2	1.085	109.035	62.420		
0.036813											
6	H10	hc	E	4	3	2	1.087	109.317	-62.420		
0.037854											
7	H11	hc	E	4	6	5	1.086	109.461	120.000		
0.041829											
8	C2	c3	M	4	3	2	1.529	109.461	178.348		
0.079588											
9	H12	hc	E	8	4	3	1.086	109.035	62.429	-	
0.001791											
10	H13	hc	E	8	4	3	1.088	109.317	-62.429	-	
0.000421											
LOOP											
IMPROPER											
DONE											
STOP											

Taula 6.2.1.2.3 Composició fitxer en format PREPI del monòmer PEH

0	0	2									
This is a remark line											

```

ps.res
PS   INT   0
CORRECT      OMIT DU   BEG
  0.0000
  1  DUMM  DU    M    0  -1  -2    0.000    .0    .0
.00000
  2  DUMM  DU    M    1   0  -1    1.545    .0    .0
.00000
  3  DUMM  DU    M    2   1   0    1.545  111.950    .0
.00000
  4  C2    c3    M    3   2   1    1.545  113.720  175.470
0.150623
  5  C1    ca    S    4   3   2    1.523  112.440  154.014
0.117905
  6  C3    ca    B    5   4   3    1.393  121.525 -151.028 -
0.244532
  7  C4    ca    B    6   5   4    1.383  121.121 -179.390 -
0.122472
  8  C5    ca    B    7   6   5    1.387  120.268    0.103 -
0.191193
  9  C6    ca    B    8   7   6    1.383  119.342   -0.101 -
0.122472
 10  C7    ca    S    9   8   7    1.387  120.108    0.052 -
0.244532
 11  H6    ha    E   10   9   8    1.077  119.288 -179.989
0.119776
 12  H5    ha    E    9   8   7    1.075  120.174 -179.999
0.119802
 13  H4    ha    E    8   7   6    1.076  120.299  179.971
0.125244
 14  H3    ha    E    7   6   5    1.075  119.715  179.984
0.119802
 15  H2    ha    E    6   5   4    1.077  119.897    0.591
0.119776
 16  H1    hc    E    4   3   2    1.088  107.110   36.185
0.006170
 17  C8    c3    M    4   3   2    1.540  111.530  -82.953
0.019598
 18  H7    hc    E   17   4   3    1.087  110.300   30.142
0.013253
 19  H8    hc    E   17   4   3    1.088  109.600  -85.409
0.013253

LOOP
  C7    C1

IMPROPER
  C2    C3    C1    C7
  C1    C4    C3    H2
  C3    C5    C4    H3
  C4    C6    C5    H4
  C5    C7    C6    H5
  C1    C6    C7    H6

DONE
STOP

```

Taula 6.2.1.2.4 Composició fitxer en format PREPI del monòmer PS

```

0      0      2

This is a remark line
molecule.res
HED    INT    0
CORRECT      OMIT DU    BEG
0.0000
1  DUMM  DU    M    0  -1  -2    0.000    .0    .0
.00000
2  DUMM  DU    M    1    0  -1    1.449    .0    .0
.00000
3  DUMM  DU    M    2    1    0    1.523   111.21    .0
.00000
4  CHE   c3    M    3    2    1    1.540   111.208  -180.000 -
0.270208
5  HH1   hc    E    4    3    2    1.079   153.630   -6.828
0.104175
6  HH2   hc    E    4    3    2    1.085    97.202  -176.271
0.068481
7  HH3   hc    E    4    3    2    1.084    67.425   77.005
0.097553

LOOP

IMPROPER
HH1    HH2    CHE    HH3

DONE
STOP

```

Taula 6.2.1.2.5 Composició fitxer en format PREPI del monòmer utilitzat com a cap en el sistema de Poliester-amida

```

molecule.res
FP0    INT    0
CORRECT      OMIT DU    BEG
0.0000
1  DUMM  DU    M    0  -1  -2    0.000    .0    .0
.00000
2  DUMM  DU    M    1    0  -1    1.449    .0    .0
.00000
3  DUMM  DU    M    2    1    0    1.523   111.21    .0
.00000
4  C1    c    M    3    2    1    1.540   111.208  -180.000
0.534953
5  O2    o    E    4    3    2    1.196    54.655   -64.958 -
0.516989
6  C3    c3    M    4    3    2    1.518    96.088   59.681 -
0.009561
7  H4    hc    E    6    4    3    1.082   109.577   156.722
0.017446
8  H61   hc    E    6    4    3    1.089   106.936   41.105
0.015316

```

9	C5	c3	M	6	4	3	1.522	112.719	-79.753
0.062584									
10	H6	hc	E	9	6	4	1.087	109.866	177.050
0.026778									
11	H62	hc	E	9	6	4	1.081	111.252	58.846 -
0.020502									
12	C7	c	M	9	6	4	1.521	111.003	-64.090
0.271229									
13	O8	o	E	12	9	6	1.204	121.457	-16.101 -
0.515313									
14	N9	n	M	12	9	6	1.351	115.001	166.300 -
0.030556									
15	H10	hn	E	14	12	9	0.999	117.723	-13.906
0.053759									
16	C11	c3	M	14	12	9	1.443	121.644	-175.202
0.103866									
17	C13	c3	3	16	14	12	1.538	113.418	-118.521 -
0.041694									
18	C15	ca	S	17	16	14	1.515	116.539	-57.398
0.017345									
19	C16	ca	B	18	17	16	1.391	120.976	-98.150 -
0.082304									
20	C18	ca	B	19	18	17	1.387	120.818	-177.832 -
0.239126									
21	C20	ca	B	20	19	18	1.386	120.290	-0.489 -
0.086937									
22	C22	ca	B	21	20	19	1.386	119.455	0.358 -
0.181956									
23	C23	ca	S	22	21	20	1.386	120.081	0.215 -
0.109115									
24	H24	ha	E	23	22	21	1.075	119.470	178.800
0.096259									
25	H25	ha	E	22	21	20	1.076	120.124	179.426
0.148497									
26	H21	ha	E	21	20	19	1.076	120.287	179.795
0.128312									
27	H19	ha	E	20	19	18	1.075	119.715	179.153
0.173652									
28	H17	ha	E	19	18	17	1.074	119.389	2.027
0.131152									
29	H26	hc	E	17	16	14	1.083	107.853	178.895
0.028908									
30	H14	hc	E	17	16	14	1.085	106.299	64.311
0.029922									
31	H12	h1	E	16	14	12	1.083	105.849	-1.496
0.033168									
32	C27	c	M	16	14	12	1.527	112.566	111.496
0.729601									
33	O28	o	E	32	16	14	1.190	123.803	153.044 -
0.586183									
34	O29	os	M	32	16	14	1.317	112.111	-31.619 -
0.327837									
35	C30	c3	M	34	32	16	1.428	119.161	-168.097
0.018156									
36	H31	h1	E	35	34	32	1.081	108.301	43.956
0.075465									
37	H32	h1	E	35	34	32	1.082	109.193	-73.661

0.071510									
38	C33	c3	M	35	34	32	1.523	107.473	163.711
0.034994									
39	H34	hc	E	38	35	34	1.085	107.408	-43.756
0.020489									
40	H63	hc	E	38	35	34	1.086	107.706	-158.412
0.013693									
41	C35	c3	M	38	35	34	1.536	117.242	79.881 -
0.014489									
42	H36	hc	E	41	38	35	1.085	107.287	-169.690
0.019881									
43	H64	hc	E	41	38	35	1.087	110.246	74.698
0.027691									
44	C37	c3	M	41	38	35	1.525	116.551	-48.609
0.065424									
45	H38	h1	E	44	41	38	1.072	112.512	-28.458
0.083276									
46	H39	h1	E	44	41	38	1.081	110.770	-150.728
0.089767									
47	O40	os	M	44	41	38	1.438	108.038	93.387 -
0.380472									
48	C41	c	M	47	44	41	1.310	120.870	-154.030
0.526635									
49	O42	o	E	48	47	44	1.198	124.742	11.919 -
0.417928									
50	C43	c3	M	48	47	44	1.530	112.901	-167.690
0.082769									
51	C44	c3	3	50	48	47	1.539	109.169	118.629 -
0.071560									
52	C46	ca	S	51	50	48	1.515	114.069	-60.510
0.009433									
53	C54	ca	B	52	51	50	1.392	120.412	-81.345 -
0.186490									
54	C53	ca	B	53	52	51	1.385	120.991	179.144 -
0.149626									
55	C51	ca	B	54	53	52	1.387	120.084	0.107 -
0.105172									
56	C49	ca	B	55	54	53	1.384	119.498	0.054 -
0.194770									
57	C47	ca	S	56	55	54	1.387	120.186	-0.056 -
0.063083									
58	H48	ha	E	57	56	55	1.076	119.612	179.956
0.108837									
59	H50	ha	E	56	55	54	1.076	120.068	179.693
0.147685									
60	H52	ha	E	55	54	53	1.075	120.233	179.834
0.131050									
61	H56	ha	E	54	53	52	1.076	119.847	179.804
0.141498									
62	H55	ha	E	53	52	51	1.077	119.536	-1.186
0.152148									
63	H57	hc	E	51	50	48	1.084	106.541	177.574
0.067655									
64	H45	hc	E	51	50	48	1.084	108.366	62.458
0.050107									
65	H58	h1	E	50	48	47	1.082	107.724	0.123
0.112691									


```

66 N59 n M 50 48 47 1.445 113.112 -119.302 -
0.635566
67 H60 hn E 66 50 48 1.002 110.377 -55.082
0.313628

LOOP
C23 C15
C47 C46

IMPROPER
-M C3 C1 O2
C5 N9 C7 O8
C7 C11 N9 H10
C13 C16 C15 C23
C15 C18 C16 H17
C16 C20 C18 H19
C18 C22 C20 H21
C20 C23 C22 H25
C15 C22 C23 H24
C11 O28 C27 O29
C43 O42 C41 O40
C44 C54 C46 C47
C46 C53 C54 H55
C54 C51 C53 H56
C53 C49 C51 H52
C51 C47 C49 H50
C46 C49 C47 H48

DONE
STOP

```

Taula 6.2.1.2.6 Composició fitxer en format PREPI del monòmer Poliester-amida centre

```

0 0 2

This is a remark line
molecule.res
TAI INT 0
CORRECT OMIT DU BEG
0.0000
1 DUMM DU M 0 -1 -2 0.000 .0 .0
.00000
2 DUMM DU M 1 0 -1 1.449 .0 .0
.00000
3 DUMM DU M 2 1 0 1.523 111.21 .0
.00000
4 CTE c3 M 3 2 1 1.540 111.208 -180.000 -
0.469368
5 HT1 hc E 4 3 2 1.085 150.547 55.923
0.149663
6 HT2 hc E 4 3 2 1.080 50.146 2.766
0.168928
7 HT3 hc E 4 3 2 1.085 99.869 -104.296
0.150777

```

```

LOOP

IMPROPER
    HT1    HT2    CTE    HT3

DONE
STOP

```

Taula 6.2.1.2.7 Composició fitxer en format PREPI del monòmer utilitzat com a cua en el sistema de Poliester-amida

6.2.2. Anàlisi dels resultats

Per analitzar els resultats mostrarem els fitxers que dóna el programa en format PDB, visualitzarem gràficament els sistemes amb un visor de PDB; i comprovarem que les dades són coherents.

6.2.2.1. Fitxers en format PDB

A continuació es mostra un exemple dels fitxers PDB resultants on es pot observar les posicions de cada un dels àtoms que forma el sistema en coordenades cartesianes.

Es mostra part del contingut dels fitxers PDBs resultants i serà en el Annex4 on es podran visualitzar el contingut original de tots els fitxers generats, tant els que s'ha aplicat les condicions periòdiques de contorn com els que es mostren com a molècula única.

ATOM	1	C1	PEH	A	0	2.467	0.022	0.595	1.00	0.00	C
ATOM	2	H9	PEH	A	0	3.027	-0.526	-0.155	1.00	0.00	H
ATOM	3	H10	PEH	A	0	2.588	-0.369	1.602	1.00	0.00	H
ATOM	4	H11	PEH	A	0	1.419	-0.023	0.313	1.00	0.00	H
ATOM	5	C2	PEH	A	0	2.987	1.459	0.555	1.00	0.00	C
ATOM	6	H12	PEH	A	0	2.453	2.049	1.294	1.00	0.00	H
ATOM	7	H13	PEH	A	0	2.900	1.845	-0.459	1.00	0.00	H
ATOM	8	C3	PE	A	1	4.458	1.481	0.969	1.00	0.00	C
ATOM	9	H14	PE	A	1	5.034	0.899	0.252	1.00	0.00	H
ATOM	10	H15	PE	A	1	4.531	1.125	1.995	1.00	0.00	H
ATOM	11	C4	PE	A	1	4.989	2.912	0.889	1.00	0.00	C
ATOM	12	H16	PE	A	1	4.437	3.532	1.593	1.00	0.00	H

ATOM	13	H17	PE	A	1	4.925	3.245	-0.146	1.00	0.00	H
ATOM	14	C3	PE	A	2	6.449	2.941	1.342	1.00	0.00	C
ATOM	15	H14	PE	A	2	7.040	2.329	0.664	1.00	0.00	H

Taula 6.2.2.1.1 Fitxer PDB PEH-23PE-PET sense coordenades periòdiques

ATOM	1	C1	PEH	A	0	2.723	0.075	-0.019	1.00	0.00	C
ATOM	2	H9	PEH	A	0	3.082	-0.401	-0.926	1.00	0.00	H
ATOM	3	H10	PEH	A	0	3.079	-0.398	0.893	1.00	0.00	H
ATOM	4	H11	PEH	A	0	1.638	0.043	-0.044	1.00	0.00	H
ATOM	5	C2	PEH	A	0	3.233	1.516	-0.061	1.00	0.00	C
ATOM	6	H12	PEH	A	0	2.899	2.036	0.832	1.00	0.00	H
ATOM	7	H13	PEH	A	0	2.909	1.986	-0.987	1.00	0.00	H
ATOM	8	C3	PE	A	1	4.761	1.517	-0.012	1.00	0.00	C
ATOM	9	H14	PE	A	1	5.142	1.005	-0.893	1.00	0.00	H
ATOM	10	H15	PE	A	1	5.075	1.076	0.933	1.00	0.00	H
ATOM	11	C4	PE	A	1	5.272	2.956	-0.095	1.00	0.00	C
ATOM	12	H16	PE	A	1	4.912	3.508	0.770	1.00	0.00	H
ATOM	13	H17	PE	A	1	4.965	3.375	-1.052	1.00	0.00	H
ATOM	14	C3	PE	A	2	6.798	2.961	-0.005	1.00	0.00	C
ATOM	15	H14	PE	A	2	7.203	2.415	-0.855	1.00	0.00	H
...											

Taula 6.2.2.1.3 Fitxer PDB PEH-23PE-PET amb coordenades periòdiques

ATOM	1	C1	PEH	A	0	1.553	0.941	2.589	1.00	0.00	C
ATOM	2	H9	PEH	A	0	2.413	0.287	2.684	1.00	0.00	H
ATOM	3	H10	PEH	A	0	0.840	0.845	3.403	1.00	0.00	H
ATOM	4	H11	PEH	A	0	1.064	0.699	1.650	1.00	0.00	H

ATOM	5	C2	PEH	A	0	2.098	2.370	2.573	1.00	0.00	C
ATOM	6	H12	PEH	A	0	1.269	3.064	2.480	1.00	0.00	H
ATOM	7	H13	PEH	A	0	2.838	2.463	1.780	1.00	0.00	H
ATOM	8	C2	PS	A	1	2.735	2.807	3.911	1.00	0.00	C
ATOM	9	C1	PS	A	1	3.785	3.894	3.727	1.00	0.00	C
ATOM	10	C3	PS	A	1	4.861	4.024	4.603	1.00	0.00	C
ATOM	11	C4	PS	A	1	5.803	5.022	4.435	1.00	0.00	C
ATOM	12	C5	PS	A	1	5.691	5.920	3.384	1.00	0.00	C
ATOM	13	C6	PS	A	1	4.629	5.806	2.505	1.00	0.00	C
ATOM	14	C7	PS	A	1	3.688	4.802	2.678	1.00	0.00	C
ATOM	15	H6	PS	A	1	2.866	4.725	1.986	1.00	0.00	H
...											

Taula 6.2.2.1.3 Fitxer PDB PEH-22PS-PET sense coordenades periòdiques

ATOM	1	C1	PEH	A	0	3.051	0.531	1.000	1.00	0.00	C
ATOM	2	H9	PEH	A	0	3.682	-0.042	0.330	1.00	0.00	H
ATOM	3	H10	PEH	A	0	3.069	0.176	2.027	1.00	0.00	H
ATOM	4	H11	PEH	A	0	2.036	0.474	0.616	1.00	0.00	H
ATOM	5	C2	PEH	A	0	3.574	1.967	0.961	1.00	0.00	C
ATOM	6	H12	PEH	A	0	2.970	2.582	1.622	1.00	0.00	H
ATOM	7	H13	PEH	A	0	3.590	2.317	-0.069	1.00	0.00	H
ATOM	8	C2	PS	A	1	4.984	2.131	1.571	1.00	0.00	C
ATOM	9	C1	PS	A	1	5.732	3.325	0.993	1.00	0.00	C
ATOM	10	C3	PS	A	1	7.122	3.343	0.910	1.00	0.00	C
ATOM	11	C4	PS	A	1	7.794	4.437	0.395	1.00	0.00	C
ATOM	12	C5	PS	A	1	7.087	5.545	-0.049	1.00	0.00	C

ATOM	13	C6	PS	A	1	5.706	5.544	0.026	1.00	0.00	C
ATOM	14	C7	PS	A	1	5.039	4.443	0.543	1.00	0.00	C
ATOM	15	H6	PS	A	1	3.964	4.456	0.595	1.00	0.00	H
...											

Taula 6.2.2.1.4 Fitxer PDB PEH-22PS-PET amb coordenades periòdiques

ATOM	1	CHE	HED	A	0	0.712	2.038	-2.882	1.00	0.00	C
ATOM	2	HH1	HED	A	0	0.706	1.940	-3.956	1.00	0.00	H
ATOM	3	HH2	HED	A	0	1.164	2.990	-2.622	1.00	0.00	H
ATOM	4	HH3	HED	A	0	-0.320	2.052	-2.550	1.00	0.00	H
ATOM	5	C1	FPB	A	1	1.269	3.474	-2.882	1.00	0.00	C
ATOM	6	O2	FPB	A	1	0.205	3.145	-2.446	1.00	0.00	O
ATOM	7	C3	FPB	A	1	2.560	3.146	-2.153	1.00	0.00	C
ATOM	8	H4	FPB	A	1	3.305	3.896	-2.380	1.00	0.00	H
ATOM	9	H61	FPB	A	1	2.923	2.203	-2.561	1.00	0.00	H
ATOM	10	C5	FPB	A	1	2.370	3.020	-0.648	1.00	0.00	C
ATOM	11	H6	FPB	A	1	3.308	2.734	-0.180	1.00	0.00	H
ATOM	12	H62	FPB	A	1	1.651	2.250	-0.407	1.00	0.00	H
ATOM	13	C7	FPB	A	1	1.939	4.346	-0.041	1.00	0.00	C
ATOM	14	O8	FPB	A	1	2.058	5.381	-0.643	1.00	0.00	O
ATOM	15	N9	FPB	A	1	1.469	4.268	1.224	1.00	0.00	N
...											

Taula 6.2.2.1.5 Fitxer PDB Poliesteramida sense coordenades periòdiques

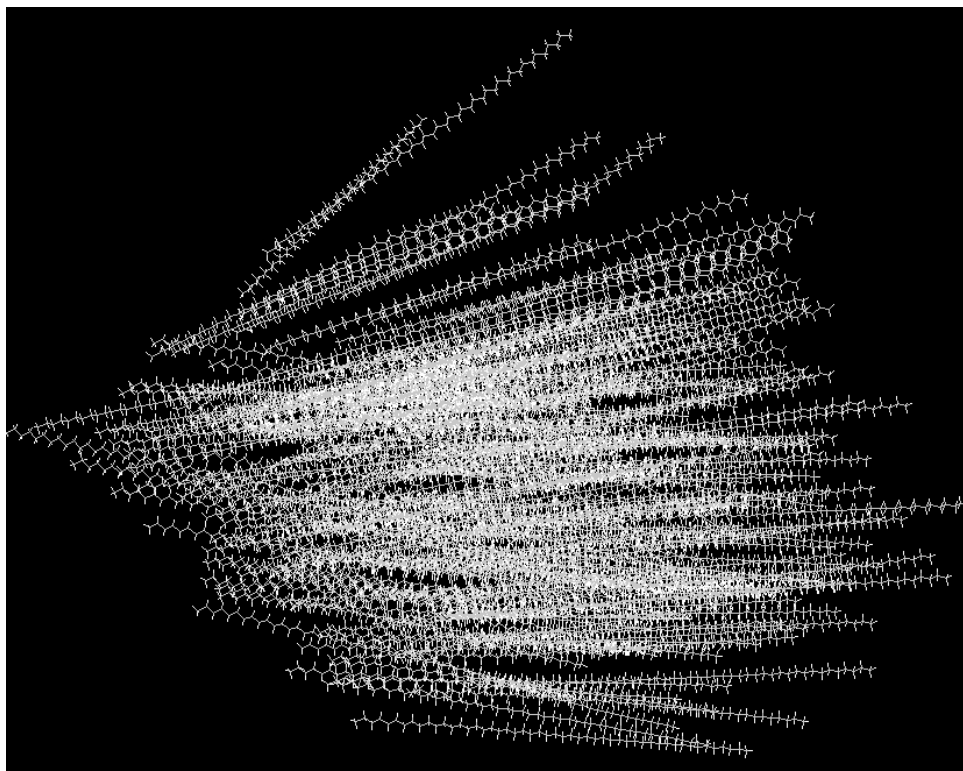
ATOM	1	CHE	HED	A	0	0.617	1.944	-2.881	1.00	0.00	C
ATOM	2	HH1	HED	A	0	0.591	1.854	-3.956	1.00	0.00	H
ATOM	3	HH2	HED	A	0	1.073	2.894	-2.623	1.00	0.00	H

ATOM	4	HH3	HED	A	0	-0.409	1.956	-2.531	1.00	0.00	H
ATOM	5	C1	FPB	A	1	1.174	3.380	-2.881	1.00	0.00	C
ATOM	6	O2	FPB	A	1	0.118	3.048	-2.429	1.00	0.00	O
ATOM	7	C3	FPB	A	1	2.477	3.047	-2.178	1.00	0.00	C
ATOM	8	H4	FPB	A	1	3.218	3.799	-2.413	1.00	0.00	H
ATOM	9	H61	FPB	A	1	2.833	2.108	-2.598	1.00	0.00	H
ATOM	10	C5	FPB	A	1	2.314	2.911	-0.671	1.00	0.00	C
ATOM	11	H6	FPB	A	1	3.261	2.622	-0.221	1.00	0.00	H
ATOM	12	H62	FPB	A	1	1.599	2.139	-0.422	1.00	0.00	H
ATOM	13	C7	FPB	A	1	1.894	4.232	-0.046	1.00	0.00	C
ATOM	14	O8	FPB	A	1	2.002	5.272	-0.644	1.00	0.00	O
ATOM	15	N9	FPB	A	1	1.447	4.146	1.226	1.00	0.00	N
...											

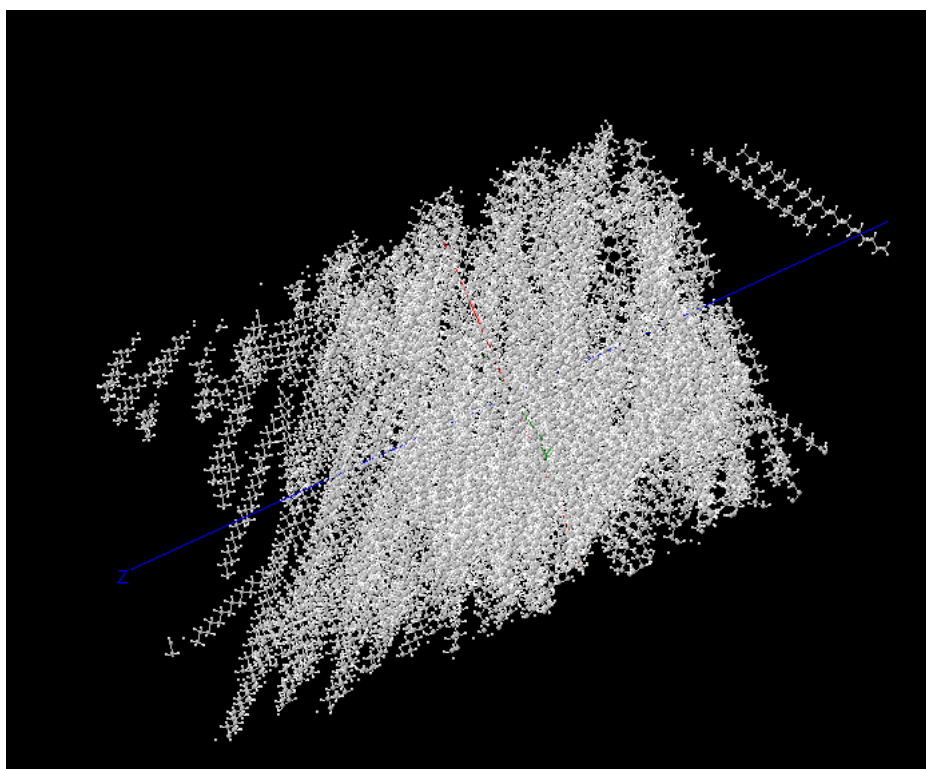
Taula 6.2.2.1.6 Fitxer PDB Poliesteramida amb coordenades periòdiques

6.2.2.2. Visualització sistema

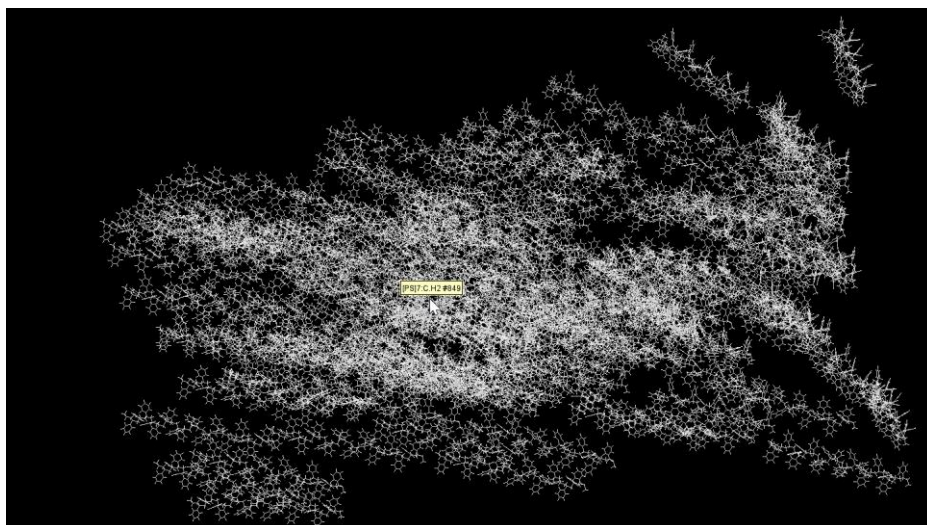
S'utilitza una aplicació anomenada "Jmol" per poder visualitzar els fitxers amb els sistemes en format PDB. Com per a cada sistema s'ha creat un fitxer PDB amb condicions periòdiques de contorn dins d'una caixa; i l'equivalent sense trobar-se dins de la caixa, tenim dues representacions diferents per cada un.



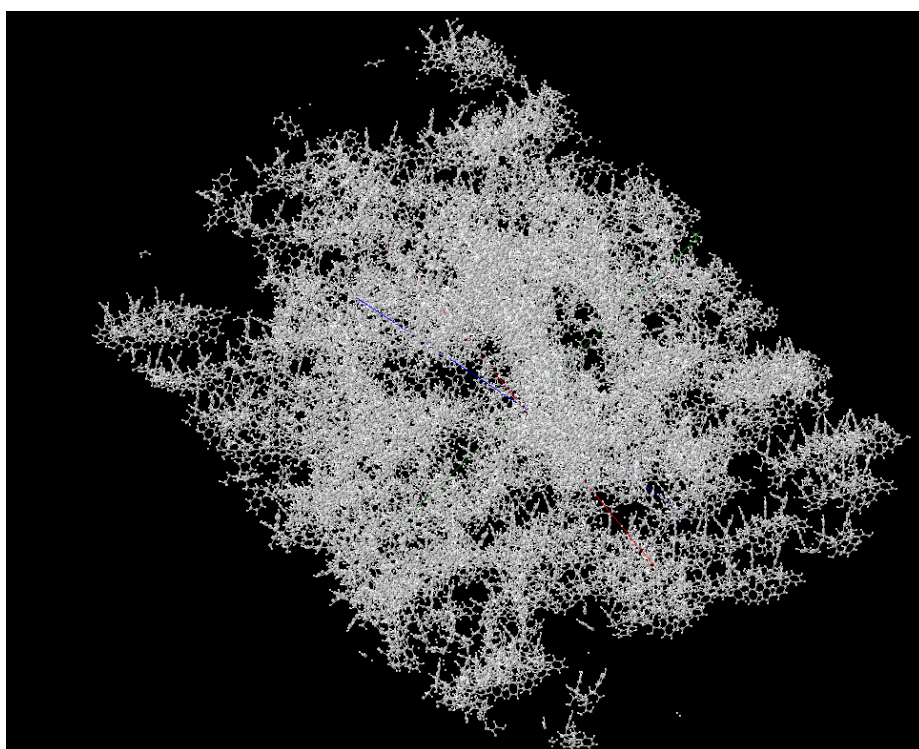
Imatge 6.2.2.2.1 Fitxer PDB PEH-23PE-PET sense coordenades periòdiques



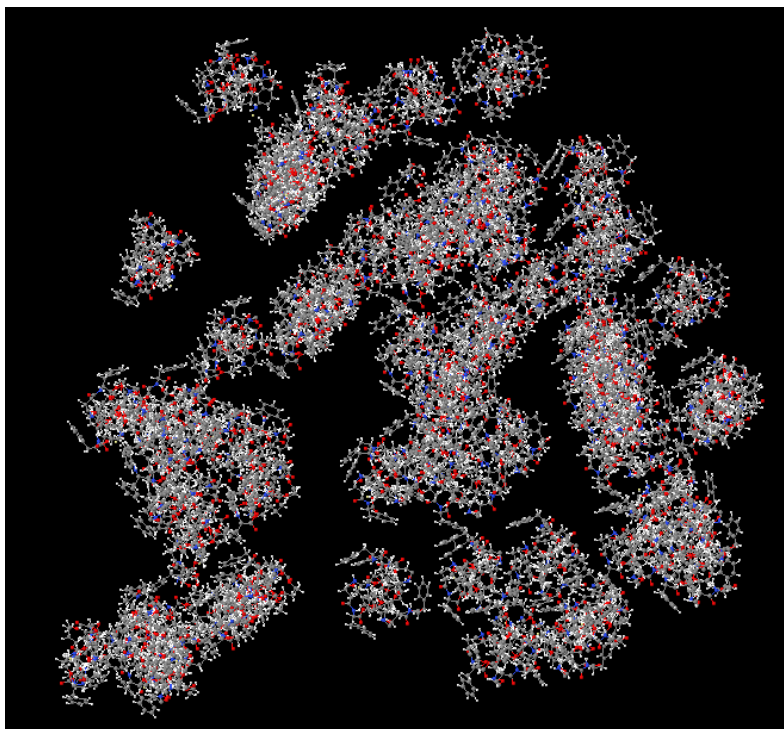
Imatge 6.2.2.2.2 Fitxer PDB PEH-23PE-PET amb coordenades periòdiques



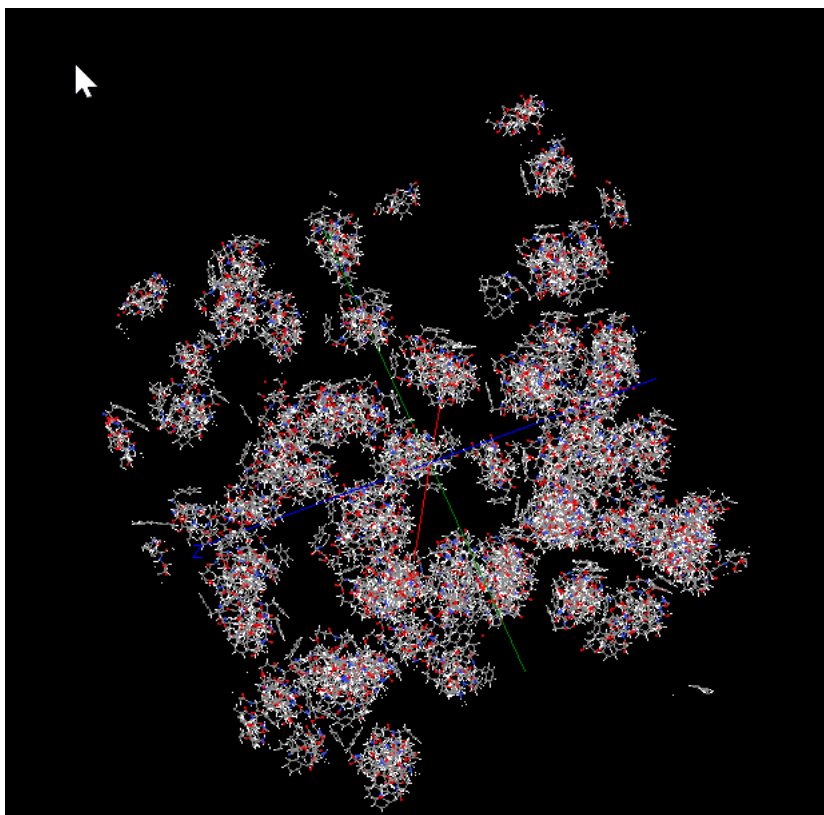
Imatge 6.2.2.2.3 Fitxer PDB PEH-22PS-PET sense coordenades periòdiques



Imatge 6.2.2.2.4 Fitxer PDB PEH-22PS-PET amb coordenades periòdiques



Imatge 6.2.2.2.5 Fitxer PDB Poliesteramida sense coordenades periòdiques



Imatge 6.2.2.2.6 Fitxer PDB Poliesteramida amb coordenades periòdiques

6.2.3. Simulació de la densitat.

Tenint en compte que la densitat experimental dels polímers estudiats és la següent^[39]:

- Polietilè $\rho_{PET} = 0,910 - 0,940 \text{ g/cm}^3$
- Poliestirè $\rho_{PES} = 1,05 \text{ g/cm}^3$
- Poliesteramida $\rho_{PEA} = 1,20 - 1,302 \text{ g/cm}^3$

Tot i que la densitat de la Poliesteramida no està reportada; es pot estimar que es troba entre els límits proposats, ja que s'han buscat densitats de polímers amb fórmula química similar com ara són:

- Poly(2,2,2'-trimethylhexamethylene terephthalamide) $\rho = 1,20 \text{ g/cm}^3$
- Polyacrylamide $\rho = 1,302 \text{ g/cm}^3$

Es realitzarà un càlcul aproximat per determinar el número de molècules i residus que hem d'incorporar a la caixa periòdica per tal d'aconseguir una densitat aproximada dels sistemes creats pel programa dissenyat.

Els polímers estan constituïts per un grup de cadenes, i cada una d'elles té un pes molecular diferent. Per tant, per obtenir el pes molecular d'un polímer (M) s'ha de calcular el pes de cada una de les cadenes presents en ell segons la següent fórmula: $M = X_n \cdot M_o$, on M_o es correspon al pes molecular de la unitat repetitiva i X_n la quantitat de repeticions o grau de polimerització.

Prenent com a exemple el Polietilè o PE amb la següent unitat repetitiva o UR “-CH₂-CH₂-”; es pot calcular el pes molecular de la UR (M_{PE}) com la suma del pes molecular de 2 carbonis i de 4 hidrògens; i per tant podríem fer la següent aproximació: $M_{PE} = 2 \cdot (12) + 4 \cdot (1) = 28 \text{ g/mol}$.

A partir del pes molecular de les UR expressat en mols podem obtenir la quantitat de molècules utilitzant la conversió de la constant d'Avogadre.

$$M_{PE} = 28 \text{ g/mol} \cdot 1 \text{ mol CH}_2 / 6,022045 \times 10^{23} = 4,64 \cdot 10^{-23} \text{ g/molècules PE}$$

Si es considera que la densitat dels Polietilè és la següent $\rho_{PE} = 0,857 \text{ g/cm}^3$ i la densitat és equivalent a “ $\rho = m/V$ ”; passant els cm³ a Å i els grams a molècules de CH₂ podem arribar al següent resultat de la densitat $\rho = 18,4255 \cdot 10^{-3} \text{ molècules PE/Å}^3$

Per, tant en aquest cas en una caixa d'un volum aproximat de $L_x 100 \text{ Å} \cdot L_y 50 \text{ Å} \cdot L_z 50 \text{ Å} = 250 \cdot 10^3 \text{ Å}^3$; caldria esperar una densitat mitja de 4606,375 molècules de PE; és a dir, el que equival a unes 184 cadenes de 25 molècules de PE.

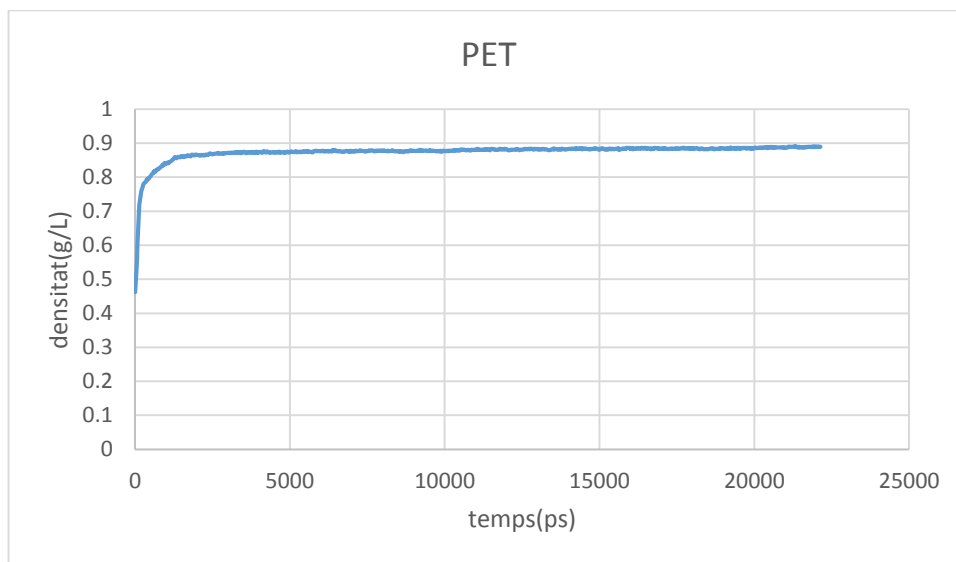
Tenint en compte també la densitat de la resta de monòmers estudiats; podem obtenir els següents resultats:

Monòmer	ρ (g/cm ³)	ρ (molècules/Å ³)	Mo(g/mol)	Mo(g/mol·cula)	Molècules V=250*10 ³ Å ³	Cadenes	UR
PE	0,910	19,57*10 ⁻³	28	4,65*10 ⁻²³	4606	184	25
PS	1,050	6,07*10 ⁻³	104,15	1,729 * 10 ⁻²³	3066	140	22
PEA	1,200	1,64*10 ⁻³	438,5	7,281*10 ⁻²²	418	84	5

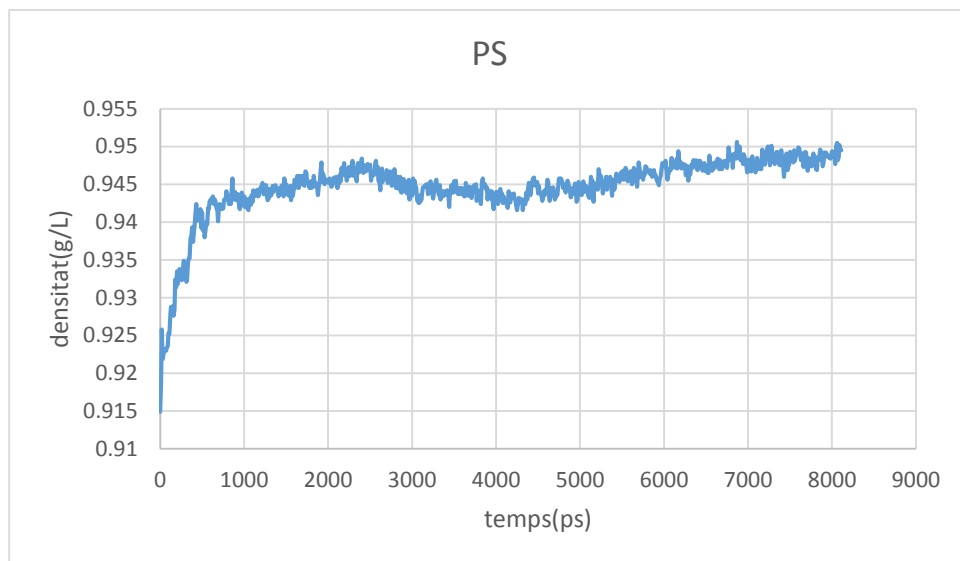
Taula 6.2.3.1 Taula amb els resultats de les densitats

Un cop generats els fitxers PDBs de sistemes amorfs dels polímers a partir dels monòmers en format PREPI; s'ha realitzat una simulació per tal de que els sistema evolucioni a pressió i pressió constants fins a assolir una densitat final constant. Per aquest motiu els fitxers PDB resultants s'han passat a membres del grup de recerca IMEM de la UPC, a partir dels quals s'han creat models atomístics que s'han relaxat utilitzant el programa AMBER^[40] de simulació per poder obtenir una densitat estable del sistema amb condicions periòdiques de contorn, a 298K i 1 atm. de pressió.

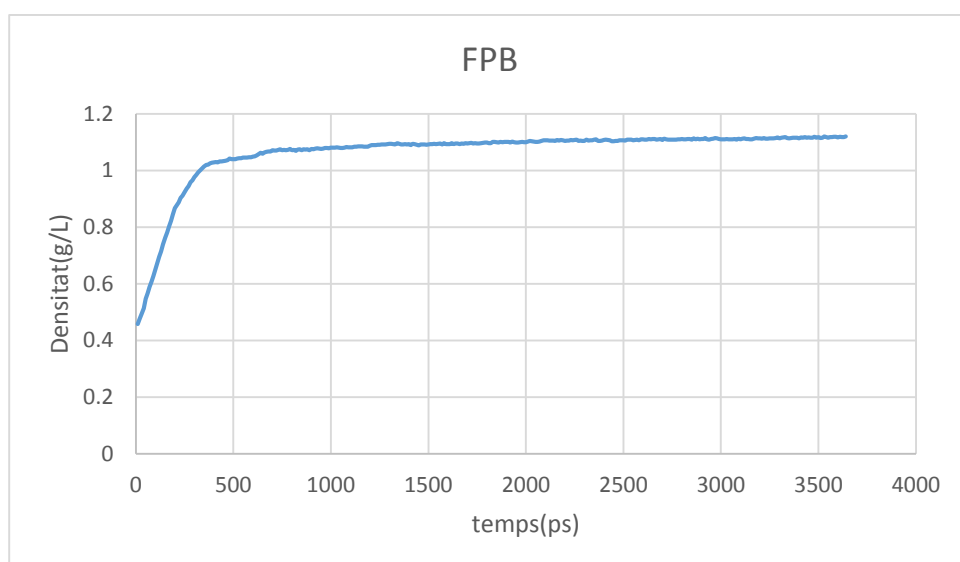
A continuació es mostren els resultats obtinguts sobre l'evolució de la densitat en funció del temps de processament per als tres polímers en forma de gràfics. Les dades amb les que s'han creat els gràfics es troben en l'Annex 4.



Gràfic 6.2.3.2 Gràfic amb l'evolució de la densitat PET



Gràfic 6.2.3.3 Gràfic amb l'evolució de la densitat PS



Gràfic 6.2.3.4 Gràfic amb l'evolució de la densitat FPB

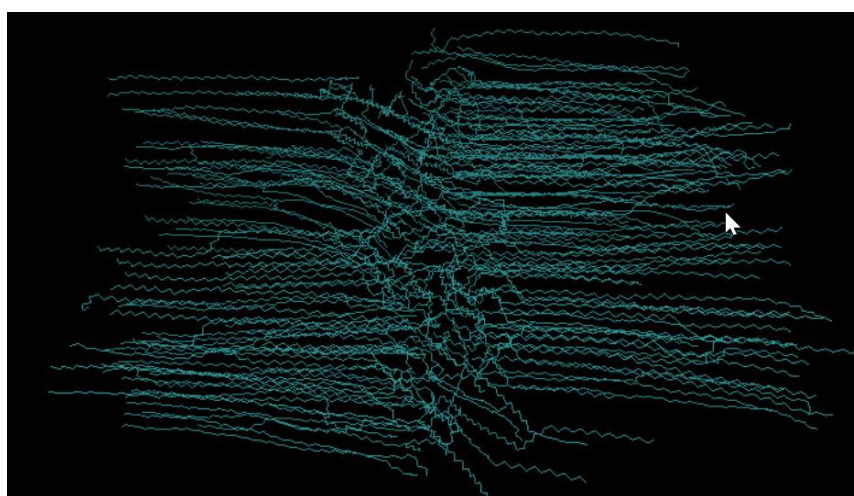
Si comparem la densitat experimental que de cada un dels polímers amb les densitats obtingudes després de la estabilització de la densitat podem comprovar que són correctes ja que sols s'ha obtingut entre un 3-10% d'error.

Monòmer	ρ (g/cm ³)	ρ inicial (g/l)	ρ final (g/l)	Temps equilibrat(ns)	% error
PE	0,910	0,462	0,890	22,140	2,21

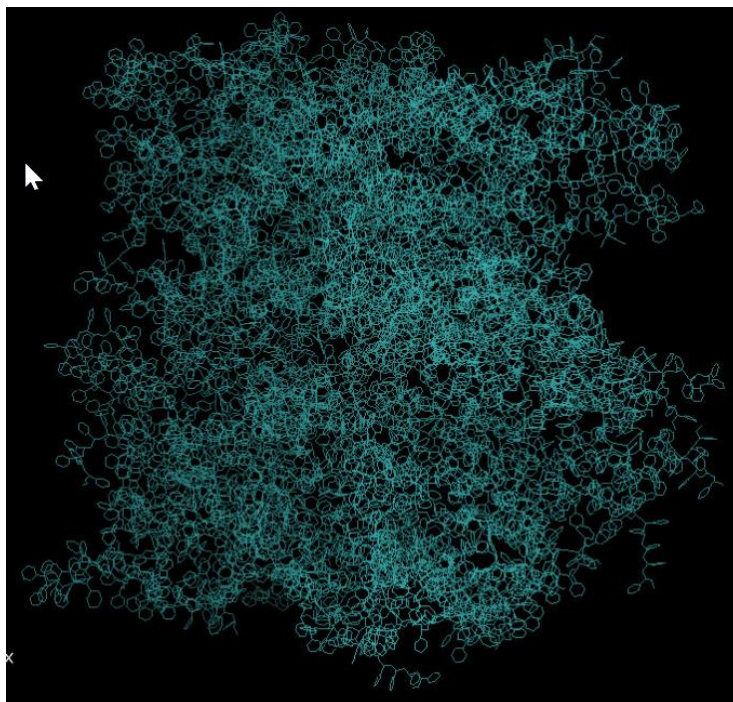
PS	1,050	0,915	0,950	8,110	9,57
PEA	1,200	0,459	1,120	3,640	6,69

Taula 6.2.3.5 Taula resum amb l'evolució de les densitats

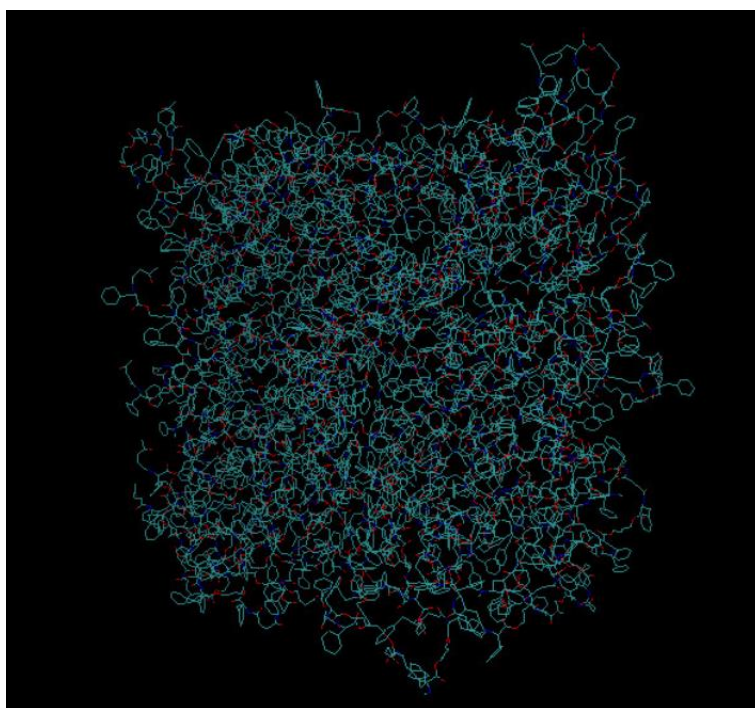
Per últim adjuntem imatges dels tres polímers, on s'han omès els àtoms d'hidrogen per donar més rellevància a l'estructura final, un cop s'ha estabilitzat la densitat:



Imatge 6.2.3.6 Sistema PET



Imatge 6.2.3.6 Sistema PS



Imatge 6.2.3.7 Sistema FPB

7. Anàlisi de l'impacte ambiental

En aquest apartat s'analitzen les conseqüències ambientals que comporta la realització d'aquest projecte.

Cal esmentar que l'impacte ambiental generat durant el seu desenvolupament és pràcticament menyspreable comparat amb treballs d'àmbit més experimental, així i tot, cal mencionar les principals fonts que causen l'impacte.

En primer lloc, referents al procés de desenvolupament del programa i la redacció de la memòria, s'han de considerar les emissions de CO₂ derivades del consum elèctric, ja que contribueixen a l'efecte hivernacle; i d'altra banda, per a la impressió i presentació de la memòria podria considerar-se la quantitat de paper utilitzada, que representa una de les principals causes de desforestació.

Emissions de CO₂ derivades del consum elèctric:

A la Taula 8.1 es mostra el detall desglossat de les emissions de CO₂ derivades del consum elèctric tenint en compte que 1kwh produeix 0,65Kg CO₂:

Element	Temps (h)	Potència (W)	Energia (KWh)	Consum CO ₂ (Kg)
Ordinador	600 h	187	112,2	72,93
Impressora	0,75 h	150	0,11	0,07
Electricitat	600 h	90	54	5,85
Total				78.85 Kg CO₂

Taula 8.1. Emissions de CO₂ derivades de consum elèctric

Residus produïts per consumibles

Es podria dir que per a la redacció de projecte s'ha utilitzat paper en format A4, tinta per impressora, CD on desar el programa i una còpia del treball, i altres materials d'oficina.

El consum total de paper ha estat aproximadament de 200 fulles en format A4, que equivalen a uns 12,46 m² de paper. Si tenim en compte que la densitat és de 80g/m², el pes equivalen és de 0,99 kg.

Conclusions

Tot i que l'aplicació desenvolupada en aquest projecte es troba encara en vies de desenvolupament, s'han acomplert els objectius plantejats en el seu inici i respon de forma satisfactòria a les especificacions que es van determinar. En general podem concloure que el projecte ha estat un èxit; ja que s'ha aconseguit l'objectiu principal; és a dir, implementar un programa informàtic que ha permès generar estructures tridimensionals de polímers d'una manera generalitzada a partir de la seva composició química.

El software que s'ha desenvolupat és una eina d'accés lliure per a qualsevol estudiant o professional que es vulguin realitzar càlculs per a la construcció de matrius polimèriques amorfes d'alta densitats, i per tant crec que és una contribució positiva, encara que petita, en aquest camp, així com a la nostra economia i societat.

En aquest primer treball s'han assentat les bases del programa que servirà per poder construir matrius polimèriques amorfes d'alta densitat. Existeixen encara molts aspectes potencials de millora que es seguiran desenvolupament per altres estudiants; i que permetran arribar a aconseguir les densitats desitjades per a les matrius que es creïn amb el programa.

Per últim s'analitzaran en detall cada una de les característiques que es van plantejar com a necessàries entre els objectius del programa:

- El programa dona com a resultats fitxers PDB que serveixen per a la construcció tridimensional de materials per simulació tenint en compte la seva estructura 3D a escala molecular.
- Amb la creació d'aquests PDBs, es simplifica la preparació de simulacions de polímers, y permet la creació d'un sistema de cadenes dels mateixos a partir de l'estructura coneguda del seus residus a partir d'unes instruccions bàsiques.
- S'ha aconseguit empaquetar tots les cadenes polimèriques dins d'una caixa rectangular ortoròmbica de tres dimensions de manera amorfa; ja que s'apliquen condicions periòdiques de contorn.
- Segons els resultats dels casos de prova un cop estabilitzada la densitat, s'ha aconseguit un marge entre 3-10% d'error si es compara la densitat experimental
- Si visualitzem les imatges dels polímers aconseguits es pot observar que no hi ha grans distorsions sobre l'estructura inicial del monòmer; ja que per encabir-lo dins de la caixa periòdica de simulació s'han tingut baixos índex de desviació.

Pressupost i/o Anàlisi Econòmica

El cost associat a l'elaboració d'aquest projecte es compon principalment en dos grans punts: despeses de mà d'obra i despeses associades a la realització.

Les despeses de mà d'obra es troben associades a les hores treballades per l'estudiant que ha realitzat el projecte tenint en compte el període d'aprenentatge en la programació en Python; i cal tenir en compte la mitja d'hores setmanals implicades durant un període aproximat de 10 mesos fins a l'entrega de la memòria escrita.

Com el treball s'ha realitzat íntegrament amb un ordinador portàtil sense requerir fase experimental, com a despeses associades a la seva realització es poden contemplar costos d'amortització de l'ordinador, consum elèctric, materials derivats de la impressió del treball i els CDs gravats amb el programa i necessaris per a la seva presentació. Evidentment, aquestes despeses seran molt menys significatives que les associades a la mà d'obra.

A la següent taula es desglossa el detall dels costos associats al projecte

DESPESES ASSOCIADAES A LA MÀ D’OBRA				
Concepte	Cost hora	Hores	Cost total	
Mà d’obra	25 €	600	15.000 €	
DESEPESES ASSOCIADAES A LA REALITZACIÓ				
Concepte	Import total	Vida útil	Amortització	Imputable
Ordinador	800 €	6 anys	10 mesos	110 €
Impressió	68 €			
CDs	2 €			
Enquadernació	6 €			
Concepte	Consum	Cost / kWh	Hores	Cost total
Electricitat	90 w	0,31	600	16,74 €
			TOTAL	15.202,74 €

Taula pressupost. Despeses associades al projecte

Bibliografia

- [1] Hal. F. Brinson, L. Catherine Brinson. *Polymer Engineering Science and Viscoelasticity*. Springer, Boston, MA. Cap 4. <https://doi-org.recursos.biblioteca.upc.edu/10.1007/978-1-4899-7485-3>
- [2] Carlos Alemán, *Modeling the tetraphenylalanine-PEG Hybrid Amphiphile: From DFT Calculations of the Peptide to Molecular Dynamics Simulations of the Conjugate*. Journal of Chemical Physics. 2011; 115(28): 8937-8946.
- [3] Carlos Alemán, *Molecular dynamics Simulation Study of Methansulfonic Acid*. Journal of Chemical Physics. 2014; 118(12): 3423-3430.
- [4] Carlos Alemán, *Influence of the Temperature on de Proton Transport Poly(syrene-co-divinylbenzene) Membranes from molecular Dynamics Simulation*. Journal of Chemical Physics. 2014; 118(21): 17643-17654.
- [5] Wei Liu, C.L. Yang, Y.T. Zhu, M.S. Wang, *Interactions between single-walled carbon nanotubes and polyethylene/polypropylene/polystyrene/poly(phenylacetylene)/poly(p-phenylenevinylene) considering repeat unit arrangements and conformations: a molecular dynamics simulation study*, Journal of Chemical Physics 2008; 112 (6): 1803–1811.
- [6] J.-L. Barrat, J. Baschnagel, A. Lyulin, *Molecular dynamics simulations of glassy polymers*, Soft Matter 2010; 6 (15): 3430–3446.
- [7] Schrödinger, LLC, *The PyMOL Molecular Graphics System*, Version 1.3r1, August, 2010.
- [8] B.P. Haley, N. Wilson, C. Li, A. Arguelles, E. Jaramillo, A. Strachan, *Polymer Modeler*, Aug., 2010. URL <https://nanohub.org/resources/9230>.
- [9] Enhanced Monte Carlo (emc), montecarlo.sourceforge.net.
- [10] Accelrys materials studio, accelrys.com.
- [11] R. F. Fedors, *Polym. Eng. Science* 1974; 14: 147 .
- [12] D. W. van Krevelen, *Properties of Polymers, Their Estimation and Correlation with Chemical Structure*, Amsterdam 1990; 3rd ed. Elsevier,
- [13] J. Bicerano, *Prediction of Polymer Properties*, New York, 1993; 2nd ed. Dekker,

- [14] P. J. Flory, *Statistical Mechanics of Chain Molecules*, Hanser, New York, 1989.
- [15] David Curco, Carlos Aleman. *Simulation of dense amorphous polymers by generating representative atomistic models*. Journal of Chemical Physics. 2003; 119(5): 2915-2922.
- [16] David Curco, Carlos Aleman. *Performance of SuSi: A Method for Generating Atomistic Models of Amorphous Polymers Based on a Random Search of Energy Minima*. Journal of Computational Chemistry. 2003; 25(6): 790-798
- [17] David Curco, Carlos Aleman. *Theoretical strategy to provide atomistic models of comblike polymers: A generation algorithm combined with configurational bias Monte Carlo*. Journal of Chemical Physics. 2004; 121(19): 9744 - 9752.
- [18] S. León, C. Alemán, F. Escalé, i M. Laso, Journal of Computational Chem, 2001; 22, 162.
- [19] S. León, D. Zanuy i C. Alemán, Journal of Computational Chem. 2002; 23, 685.
- [20] J. I. Siepmann, Mol. Phys. 1990; 70, 1145.
- [21] J. I. Siepmann and D. Frenkel, Mol. Phys 1992; 75, 59.
- [22] David Curco, Manuel Laso, Carlos Aleman. *Generation-Relaxation Algorithms to Construct Representative Atomistic Models of Amorphous Polymers: Influence of the Relaxation Method*. Journal of Computational Ch. Journal of Chemical Physics. 2004; 108(52): 20331 -20339.
- [23] Python Software Foundation 2001-2018. <https://www.python.org/>
- [24] Kindler, E.; Krivy, I. *Object-Oriented Simulation of systems with sophisticated control*. International Journal of General Systems 2011; 313–343.
- [25] Grady Booch, Ivar Jacobson i James Rumbaugh. *UML Llenguatge Unificat de Modelat*, 1994-1996 . <http://www.uml.org/>
- [26] OMG consorci americà per l'estandardització i promoció de l'Objecte Model en totes les seves formes 1989; Hewlett-Packard, IBM, Sun Microsystems, Apple Computer, American Airlines and Data General.
- [27] R. Salomon-Ferrer, D.A. Case, R.C. Walker. *An overview of the Amber biomolecular simulation package*. WIREs Comput. Mol. Sci. 2013; 3, 198-210. <http://ambermd.org>
- [28] H. M. Berman, K. Henrick, H. Nakamura. *Announcing the worldwide Protein Data Bank*. Nat Struct Biol 2003; 10: 980. . <https://www.rcsb.org/>

- [29] Jerod Parsons, J.Bradley Holmes, J.Maurice Rojas i Jerry Tsal. Practical Conversion from Torsion Space to Cartesian Space for In Silico Protein Synthesis. J Comput Chem. 2005; 26 1063-1068
- [30] The SciPy community 2008-2018. <https://www.scipy.org/>
- [31] NumPy developers 2018. <http://www.numpy.org/>
- [32] <https://docs.python.org/2/library/sys.html>
- [33] <https://docs.python.org/2/library/pdb.html>
- [34] <https://docs.python.org/2/library/os.html>
- [35] JetBrains s.r.o. Developed with drive and IntelliJ IDEA. https 2000–2018. www.jetbrains.com
- [36] Jmol -11.8 Minnesota Supercomputer Center <http://www.jmol.org/>
- [37] Scott Chacon i Ben Straub. *Pro Git* book, 2nd Edition (2014) <https://git-scm.com/>
- [38] TortoiseGit team 2.6.0 February 2018; <https://tortoisegit.org/>
- [39] Frank. W Harrys. *Introduction to Polymer Chemistry*. Journal of Chemical Education 1981;54(11)
- [40] Kai Guo, C.C.Chi, E. Chkhaidze, R.Katsarava, *Synthesis and Characterization of Novel Biodegradable Unsaturated Poly(ester amide)s* Journal of Polymer Science 2005; 43 1463-1477.
- [41] Densitats extrems de <http://scientificpolymer.com/density-of-polymers-by-density/>
- [42] Referències del programa AMBER: D.A. Case, R.M. Betz, D.S. Cerutti, T.E. Cheatham, III, T.A. Darden, R.E. Duke, T.J. Giese, H. Gohlke, A.W. Goetz, N. Homeyer, S. Izadi, P. Janowski, J. Kaus, A. Kovalenko, T.S. Lee, S. LeGrand, P. Li, C. Lin, T. Luchko, R. Luo, B. Madej, D. Mermelstein, K.M. Merz, G. Monard, H. Nguyen, H.T. Nguyen, I. Omelyan, A. Onufriev, D.R. Roe, A. Roitberg, C. Sagui, C.L. Simmerling, W.M. Botello-Smith, J. Swails, R.C. Walker, J. Wang, R.M. Wolf, X. Wu, L. Xiao and P.A. Kollman (2016), AMBER 2016, University of California, San Francisco
- [42] Dades obtingudes de la comissió europea per la producció de CO₂

Annex A1 – Manual d'usuari

El manual d'usuari que s'ha realitzat i annexat en aquest apartat per la utilització del programa conté els següents punts:

- Introducció a la funcionalitat del programa
- Requeriments previs
 - Instal·lació entorn desenvolupament de Python
 - Instal·lació classes
- Procés d'execució del programa
 - Dades d'entrada o Input
 - Modes d'execució de les classes
 - Resultats o Output amb exemples d'execució
 - Llistat de possibles errors

A1.1 Introducció

El present document és un manual complet per a la utilització d'una aplicació informàtica programada en Python per a la construcció de matrius polimèriques amorfes d'alta densitat.

El programa pretén simplificar la preparació de simulacions dinàmiques moleculars de sistemes polimèrics; ja que tot i que els polímers són sempre presents amb funcions vitals en diferents àmbits de la indústria, encara queda molt per aprendre sobre el seu comportament a nivell molecular. Per aquest motiu, els programes que permeten realitzar simulacions sobre el seu comportament molecular poden aportar molta informació sobre les seves estructures tant a nivell molecular com atòmic.

Les molècules dels polímers poden estar compostes de cadenes molt llargues que poden ser d'un únic tipus de monòmer o de diferents tipus; així com també poden existir com a barreges compostes de cadenes de diferents longituds, monòmers ramificats i diferents. Aquestes característiques són les que fan que la seva simulació sigui molt complexa fins i tot amb l'ajuda de programes informàtics.

Amb l'aplicació informàtica desenvolupada es vol facilitar la simulació de matrius polimèriques; ja que a partir de un fitxer de carrega amb les especificacions necessàries, permet la creació d'un sistema format per diferents molècules de polímers dins d'una caixa de 3D). A partir de la definició de la grandària de la caixa contenidora, així com el nombre de monòmers del sistema es pot obtenir un sistema amorf amb una densitat pròxima a la requerida. La definició de l'estructura de cada un dels

monòmers diferent es realitza per mitja d'un fitxer en format PREPI (format utilitzat per a la definició de residus de proteïnes en el programa **AMBER**¹. El resultat del programa serà la creació d'un fitxer amb les coordenades cartesianes de tots els àtoms de l'estructura del polímer en format PDB.

L'usuari ha de introduir les característiques del sistema (matrius polimèrica) que vol construir en un fitxer de carrega amb format "txt" tal i com s'especifica en els propers punts del manual. En aquest fitxer caldrà especificar:

- Nom del sistema (fitxer PDB resultant) i mida de la caixa periòdica 3D on empaquetar el sistema.
- Paràmetres per a la detecció de col·lisions: distància mínima de col·lisió, reintents de col·locació, i % màxim de pertorbació del sistema en cas de detectar error.
- Definició del sistema a generar: Numero de molècules que ha de tenir el sistema, composició de cada una de les molècules i ubicació dels fitxers en format PREPI (monòmers) que componen les molècules del sistema.

Amb el contingut del fitxer de carrega, les característiques del sistema i els fitxers en format PREPI dels diferents tipus de monòmers necessaris per compondre les molècules, el programa inicia la construcció del sistema.

El programa tenint en compte la estructura i coordenades de cada tipus de monòmer, tal i com venen especificades en els fitxers PREPI, comença a enllaçar els residus de cada molècula seguint la cadena principal segons el nombre i ordre especificat en el fitxer d'entrada. Un cop col·locat cada monòmer es revisa que en el resultat no hi hagi present col·lisions calculant la distància entre els àtoms i sempre considerant error quan es supera la distància mínima informada (col·lisió). En cas de detectar col·lisió s'intentaran tornar a recol·locar els residus amb noves orientacions dins de la caixa sempre respectant un % de pertorbació màxima sobre l'estructura de cada monòmer, i realitzant el numero de re-intents especificats.

Cal tenir en compte que el sistema es vol crear dins d'una caixa de tres dimensions (3D) amb condicions periòdiques per simular millor les propietats d'un sistema macroscòpic; i que cal considerar la caixa com si estigués envoltada de imatges o rèpliques d'ella mateixa. Això comporta que quan un àtom es trobi a prop d'una paret de la caixa o límit; en comptes de sortir fora d'ella, el programa realitzarà el càlculs necessaris per que aquest torni a aparèixer a l'altre costat de la caixa.

AMBER¹: Es pot trobar informació del programa de proteïnes a la següent url: <http://ambermd.org>

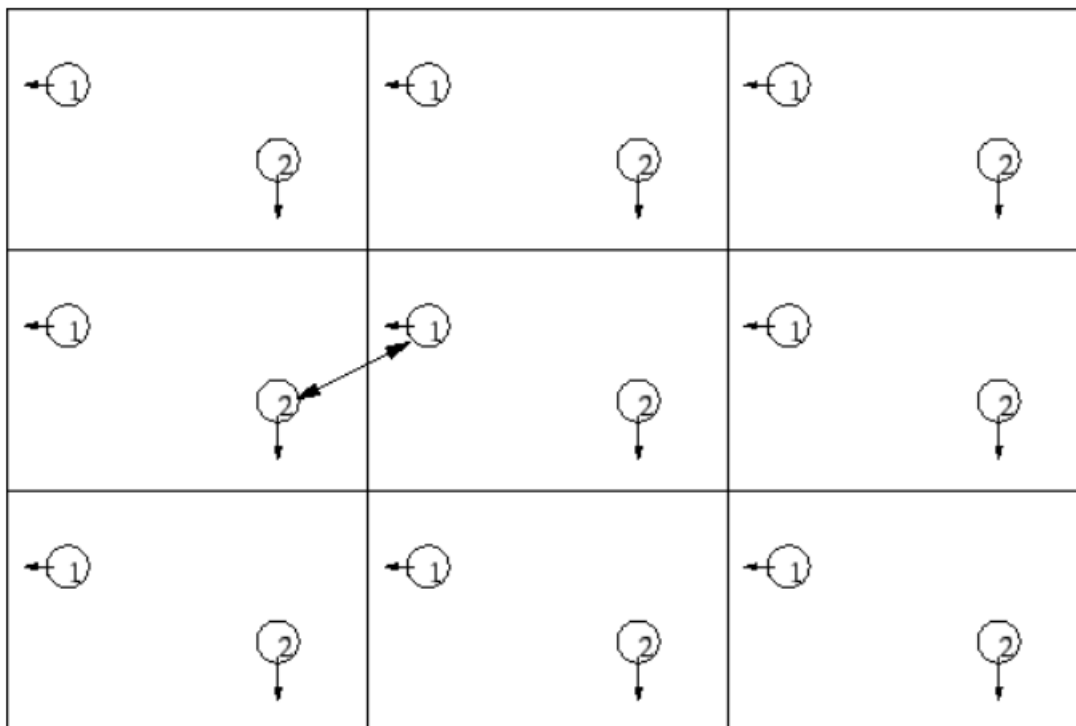


Figura A1.1.1 Exemple de condicions periòdiques en 2D: La partícula 1 està a punt de sortir de la cara esquerra de la cel·la central i entrar a la cel·la central, encara que per la cara dreta. [<https://web.northeastern.edu/afeiguin/p4840/p131spring04/node41.html>]

Finalment, un cop col·locades totes les molècules especificades en el fitxer d'entrada o de càrrega, els programa crea el fitxer PDB del sistema resultant; i en cas d'error mostra un missatge per a que l'usuari pugui realitzar les correccions necessàries en el fitxer de input.

A.1.2 Requeriments previs

A1.2.1 Instal·lació entorn desenvolupament Python

Per poder executar el programa cal tenir instal·lat la versió de Python 2.7 o superior. Aquesta versió del programa no és compatible amb les versions de Python 3.X.

<https://www.python.org/downloads/>

A banda del programa Python també s'han d'instal·lar algunes llibreries addicionals:

A1.2.1.1 Llibreries addicionals necessàries de Python

S'han d'instal·lar unes llibreries que no venen amb la instal·lació del Python 2.7. Aquestes llibreries són necessàries en cas de que no estiguin ja instal·lades.

Si no tenim instal·lat el PIP caldrà instal·lar-ho.

- Baixar el fitxer de l'adreça <https://bootstrap.pypa.io/get-pip.py>

Copiar el fitxer dintre del directori on tenim instal·lat el Python, exemple C:\Python27.

Per instal·lar el PIP, en el cas de Windows caldrà fer servir la instrucció CMD i anar al directori on tenim instal·lat el Python i escriure la instrucció: **python get-pip.py**

A continuació aniran sortint diferents missatges indicant que s'està instal·lant el PIP.

- Si no tenim instal·lada la llibreria numpy caldrà instal·lar-la.

Obrim una finestra de comandes de Windows, anem al directori C:\Python27\scripts i executem la instrucció: **pip install numpy**

A continuació aniran sortint diferents missatges indicant que s'està instal·lant la llibreria numpy.

D'aquesta forma s'instal·larà la llibreria numpy necessària per executar el programa SuSi. Aquesta acció sols cal realitzar-la una vegada.

A1.2.2 Instal·lació del programa SuSi

La instal·lació del mòduls en Python del programa SuSi es farà de la següent manera:

A1.2.2.1 Fitxer susi-1.0.0.rar

Dintre del fitxer susi-1.0.0.rar es troben tots els fitxers i directoris.

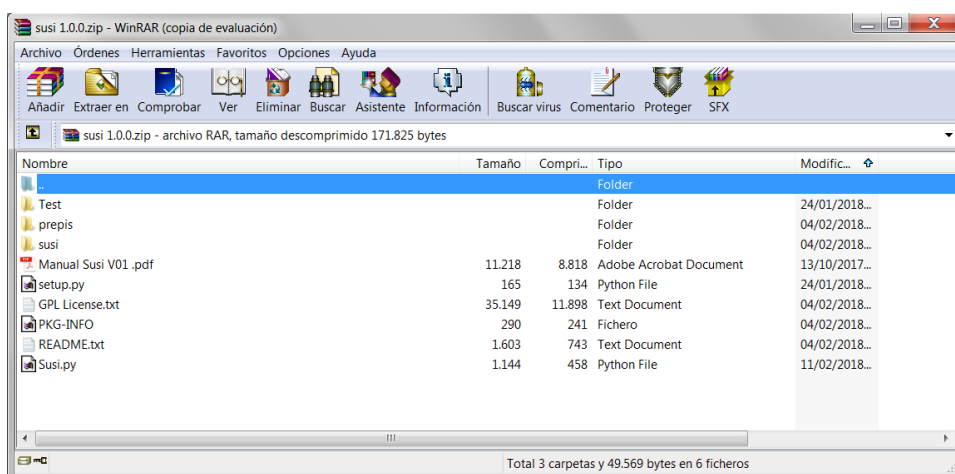


Figura A1.2.2.1.1 Contingut fitxer rar amb el instal·lable

Un cop descomprimit es mostra aquesta estructura de fitxers i directoris:

Directori, fitxer	Descripció
\test	Carpeta amb fitxers de carrega .txt que són exemples per la creació de matrius polimèriques. El contingut d'aquest directori s'haurà de moure al directori on tenim instal·lat el Python, per exemple C:\Python27
\prepis	Carpeta amb fitxers .prepi amb exemples de monòmers necessaris per la creació de matrius polimèriques. Aquest directori s'ha de moure al directori on tenim el Python, per exemple C:\Python27\prepis
\susi	Carpeta amb els següents mòduls Python: <ul style="list-style-type: none"> • __init__.py • atom.py • builer.py • filePDB.py • filePREPI.py • files.py • int2car_M.py • messages.py • molecule.py • residue.py • system.py
Manual Manual_Susi_V01.pfd	Manual utilització programa .
setup.py	Fitxer que servirà per instal·lar els mòduls de Python.
GPL License.txt	Llicència programari.
PKG-INFO	Informació relativa a la distribució dels fitxers.
Readme.txt	Fitxer amb les instruccions d'instal·lació.
Susi.py	Fitxer Python principal per a l'execució del programa SuSi que crearà el fitxer PDB

Taula A1.2.2.1.2 Taula amb el contingut del fitxer.rar amb el instal·lable

A1.2.2.2 Fitxer setup.py

Per procedir a instal·lar el sistema dels mòduls de Python, cal situar-se on es troba el fitxer setup.py i escriure aquest comandament:

“setup.py install”

Exemple pel sistema operatiu Windows:

```
C:\Python27\susi-1.0.0>setup.py install
```

Figura A1.2.2.2.1 Instruccions a executar per a realitzar la instal·lació

Depenen de la plataforma amb que s'estigui treballant, els mòduls Python s'instal·laran automàticament en uns o altres directoris.

Plataforma	Directori estàndard	Valor per defecte
Windows	\Lib\site-packages	C:\PythonXY\Lib\site-packages\susi
Unix (pure)	/lib/pythonX.Y/site-packages	/usr/local/lib/pythonX.Y/site-packages/susi
Unix (non-pure)	/lib/pythonX.Y/site-packages	/usr/local/lib/pythonX.Y/site-packages/susi

Figura A1.2.2.2.2 Taula amb els noms dels directoris i els seus valors

Un cop realitzada la instal·lació, ja es troben tots els mòduls de Python en els directoris adequats per poder començar a treballar.

Exemple amb el sistema operatiu Windows

C:\Python27\Lib\site-packages\susi













	<code>_init_.py</code>	24/01/2018 20:31	Python File	1 kB
	<code>atom.py</code>	06/02/2018 20:32	Python File	4 kB
	<code>builder.py</code>	06/02/2018 20:45	Python File	49 kB
	<code>collision.py</code>	04/02/2018 16:00	Python File	5 kB
	<code>filePDB.py</code>	06/02/2018 20:31	Python File	13 kB
	<code>filePREPI.py</code>	24/01/2018 20:31	Python File	8 kB
	<code>files.py</code>	24/01/2018 20:31	Python File	4 kB
	<code>int2car_M.py</code>	24/01/2018 20:31	Python File	5 kB
	<code>messages.py</code>	24/01/2018 20:31	Python File	3 kB
	<code>molecule.py</code>	24/01/2018 20:31	Python File	4 kB
	<code>residue.py</code>	24/01/2018 20:31	Python File	3 kB
	<code>system.py</code>	24/01/2018 20:31	Python File	4 kB

Figura A1.2.2.2.3 Imatge amb les diferents classes del directori susi

A1.2.3 Directoris i fitxers de treball

Un cop descomprimit el fitxer “rar” i feta la instal·lació del “setup”, caldrà moure alguns fitxers que tenim dintre del fitxer comprimit:

- Directori prepis

Al descomprimir el fitxer “rar” es crea un directori “prepis” on es troben emmagatzemats alguns fitxers en format PREPI (monòmers) que es poden utilitzar per anar composant molècules.

Aquest directori prepis s’ha de moure al directori on tenim el Python, per exemple dintre del directori C:\Python27\prepis

- Directori test

Dintre del directori “test” es poden localitzar diferents exemples de fitxers de càrrega amb el format correcte per la creació de sistemes.

El contingut d’aquest directori s’ha de moure on tenim instal·lat el Python, per exemple C:\Python27

- Fitxer Susi.py

Aquest fitxer s’haurà de moure al directori on tenim instal·lat el Python, per exemple C:\Python27

Per executar el programa, cal desar els fitxers “.txt” d’exemple o altres creats de nou en l’arrel del directori on s’ha realitzat la configuració.

Finalment l’execució del programa, si tot és correcte, donarà com a resultat un fitxer en format PDB en la pròpia arrel.

A continuació es descriu el que es trobarà en l’arrel del directori on s’ha realitzat la instal·lació i descompressió del fitxer RAR.

Directori	Descripció
\prepis	Directori amb els fitxers PREPI que contenen els monòmers que volem que el programa faci servir per la creació del fitxer PDB.
\test	Directori amb fitxers d’exemple de configuració del sistema
\susi	Classes python
\	<p>En aquest directori s’hi trobaran:</p> <ol style="list-style-type: none"> 1. Fitxer “README” amb les instruccions de instal·lació. 2. Susy.py fitxer necessari per l’execució del programa principal. 3. Manual d’usuari de l’aplicació. 4. Llicència. 5. Fitxer distribució. 6. Cal desar directament els fitxers de càrrega que es volen utilitzar i dels que hi ha exemples en el directori /test. 7. Un cop executat el programa, es desen els fitxers PDB si tot funciona correctament.

Taula A1.2.2.3.1 Taula amb l’estructura del directori

Exemple sistema operatiu Windows:

.idea	25/01/2018 22:03	Carpeta de fitxers	
dist	18/01/2018 23:11	Carpeta de fitxers	
prepis	14/01/2018 17:22	Carpeta de fitxers	
susi 1.0.0	26/01/2018 18:49	Carpeta de fitxers	
test	04/02/2018 18:48	Carpeta de fitxers	
1ALA.pdb	23/01/2018 22:01	Fitxer PDB	9 kB
create1ala.txt	16/01/2018 22:33	Document de text	1 kB
create10ala25.txt	05/01/2018 13:29	Document de text	1 kB
Mainbuilder.py	23/01/2018 22:01	Python File	1 kB
MANIFEST	18/01/2018 23:02	Fitxer	1 kB
Manual APLICACIÓ INFORMÀTICA PER A ...	21/01/2018 20:12	Documento de Mi...	777 kB
PKG-INFO	04/02/2018 16:41	Fitxer	1 kB
README.txt	04/02/2018 16:46	Document de text	2 kB
setup.py	16/01/2018 22:29	Python File	1 kB

Figura A1.2.2.3.2 Imatge directori en plataforma Windows

A1.3 Processament de les dades

El programa desenvolupat realitza la lectura d'un fitxer de text amb un format determinat on s'especifiquen les característiques del sistema que es vol crear (fitxer input), les ubicacions dels fitxers de lectura amb informació estructural del tots els monòmers que formen el sistema (fitxers prepri), i els fitxer de sortida

Un cop interpretades les dades i realitzada la lectura dels fitxers estructurals d'entrada, s'inicia el procés de construcció del sistema amb les seves validacions, per finalment si tot es correcte donar com a resultat el sistema desitjat en format PDB.

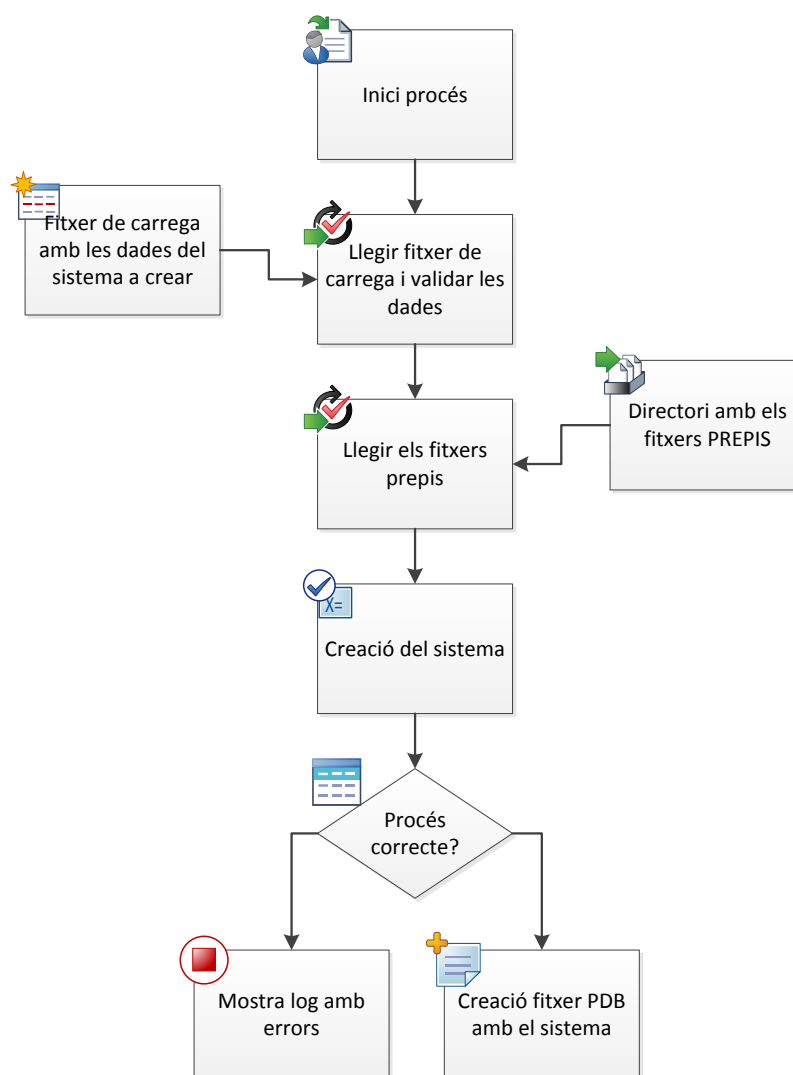


Figura A1.3.1 Flux resum procés

A1.3.1 Dades d'entrada

Les dades d'entrada necessàries per l'execució del programa són:

- Un fitxer de càrrega amb les dades del sistema.
- Un directori amb els fitxers estructurals de cada monòmer del sistema amb format PREPI.

A1.3.1.1 Fitxer de càrrega

En aquest apartat es defineix el contingut del fitxer de càrrega on s'especifiquen les característiques del sistema a crear.

Per aconseguir crear correctament el fitxer PDB caldrà indicar obligatòriament una sèrie de paràmetres. En cas d'especificar incorrectament aquest fitxer, el programa mostrarà un missatge d'error i no procedirà a la creació del fitxer PDB.

A continuació es llisten i expliquen els paràmetres que cal informar en el fitxer:

- **Nom del fitxer PD:** Indicarà el nom del fitxer PDB que es crearà si no es detecta cap error.

Dins del fitxer cal especificar el paràmetre sistema i el seu contingut "system_name, [nom]". És a dir, cal indicar el paràmetre seguit d'una coma i el nom del fitxer (sense l'extensió) que es vol crear.

Exemple: system_name,10ALA

- **Mides de la caixa periòdica 3D:** Cal informar les mesures de la caixa periòdica on es crearà el sistema sempre en Àngstroms (Å) i separats per comes.

Dins del fitxer cal especificar el paràmetre caixa i les mesures de la caixa "Box, Lx [valor] A, Ly [valor] A, Lz [valor] A". És a dir, cal indicar el paràmetre box seguit d'una coma i les mesures de la caixa en Å separades per comes.

Exemple: box,Lx 75 A,Ly 75 A,Lz 75 A

- **Coordenades periòdiques:** Indicador per especificar si el sistema creat en el PDB resultant ha de respectar les condicions periòdiques de la caixa.
 - Y – Tot el sistema es crea dintre de la caixa periòdica amb representació discontinua de les molècules tot respectant les condicions de periodicitat imposades.

- N – Tot i considerar les condicions periòdiques de contorn, el sistema es crea sense discontinuïtats en les molècules representant les imatges periòdiques fora de la cel.la unitària.

Dins del fitxer cal especificar el paràmetre per a les coordenades periòdiques i el valor de l'indicador "periodical_coordinates, [Y|N]". És a dir, cal indicar el paràmetre seguit d'una coma i el valor.

Exemple: periodical_coordinates,Y

- **Distància mínima de col·lisió:** És la mínima distància permesa entre dos àtoms no enllaçats del sistema. Aquesta distància sempre serà en Àngstroms (Å).

Dins del fitxer cal especificar el paràmetre per a la distància mínima i el valor de la distància sempre en Å "minimum_distance, [valor]". És a dir, cal indicar el paràmetre seguit d'una coma i el valor.

Exemple: minimum_distance, 1.5 Å

En cas de decimals s'utilitzarà un"." per les separacions (notació anglosaxona).

- **Re-intents:** Número de re-intents màxim que realitzarà el programa per intentar recol·locar un monòmer nou en el sistema sense que es detecti cap col·lisió, és a dir, mantenint el criteri de distància mínima.

Dins del fitxer cal especificar el paràmetre dels re-intents i el seu valor "Retries, [valor]". És a dir, cal indicar el paràmetre seguit d'una coma i el valor.

Exemple: retries, 10

- **Desviació màxima:** % de desviació dels paràmetres estructurals dels monòmers que cal tenir en compte per reubicar els nous residus. Quan es detecta una col·lisió s'introdueix una pertorbació aleatòria en les distàncies i angles d'enllaç de cada àtom del monòmer a introduir en el sistema per tal d'evitar-la. Per exemple en cas de tenir una desviació màxima de 5, es considerarà un +/-5% i per tant es calcularà entre el 0,95 i el 1,05 de desviació estructural.

Dins del fitxer cal especificar el paràmetre per informar la desviació màxima permesa i el seu valor com a percentatge (%) de desviació: "maximum_deviation, [valor]". És a dir, cal indicar el paràmetre seguit d'una coma i el valor.

Exemple: maximum_deviation, 5

- **Número de molècules:** Número de molècules que ha de tenir el sistema.

Dins del fitxer cal especificar el paràmetre per informar el número de molècules i el seu valor “molecules_number, [valor]”. És a dir, cal indicar el paràmetre seguit d’una coma i el valor.

Exemple: molecules_number,4

- **Ubicació dels fitxers PREPI:** Nom de la carpeta on es troben tots els fitxer amb format PREPI necessaris per crear el sistema. El nom dels fitxers s’utilitzarà en el fitxer de càrrega per fer referència a cada un dels monòmers necessaris per crear el sistema.

En el fitxer de càrrega caldrà especificar en un paràmetre la ubicació de la carpeta on s’han desat els fitxers PREPIs. El nom de la variable del fitxer de càrrega on cal especificar la ruta és “path_prepi, [camí]”. És a dir, cal indicar el paràmetre seguit d’una coma i el valor.

Exemple: path_prepi,prepis

- **Descripció de la composició de les molècules del sistema:** Al final del fitxer cal informar de la composició de cada molècula. S’han d’especificar tantes files com molècules ha de tenir el sistema.

El programa validarà que en el fitxer input s’hagin informat tantes files amb monòmers com número de molècules s’ha indicat en el paràmetre “molecules_number”.

En cada fila s’informarà el número de residu i el seu nom, **que ha de coincidir amb el nom del fitxer prepi** emmagatzemat en la carpeta indicat en el paràmetre “path_prepi”.

El sistema tindrà en compte el nombre i l’ordre per compondre el sistema. Per exemple: “2 ALA, 3 GLY, 1 ALA” crearà una molècula que començarà concatenant 2 monòmers ALA, seguits de 3 GLY per finalitzar amb un ALA.

Dins del fitxer cal especificar la composició de cada una de les molècules, tenint en compte que cada fila es correspon a una molècula i que davant del residu s’indica la quantitat que cal crear: “[Num1 PREPI1] [, [Num2 PREPI2] ...]”. És a dir, per exemple 2 molècules serien 2 files diferents, la primera amb 10 monòmers d’ALA seguit de 5 GLY i un ALA per tancar, mentre que la segona molècula serà de 10 ALA:

10 ALA, 5 GLY, 1 ALA

10 ALA

A1.3.1.2 Fitxer estructurals, PREPI

La construcció del sistema es realitzarà a partir de residus o monòmers que cal disposar com a fitxers d'entrada en format PREPI; ja que un residu o monòmer és la unitat molecular bàsica del programa. Els fitxers en format PREPI són els que normalment s'utilitzen per a descriure l'estructura de residus peptídics (amb coordenades internes) en el programa **AMBER²** alhora de construir proteïnes.

A continuació es mostra un exemple de fitxer PHE.prepi amb format PREPI de l'aminoàcid fenil alanina, amb les especificacions de cada una de les columnes de componen el fitxer i per a que s'utilitzen en el programa:

```

0      0      1

PHENYLALANINE PREP INPUT EXAMPLE (title)
PHE
PHE INT      1
CORRECT OMIT DU  BEG
0.0
1 DUMM DU M 0 -1 -2 0.0000 0.0000 0.0000 0.000
2 DUMM DU M 1 0 -1 1.4490 0.0000 0.0000 0.000
3 DUMM DU M 2 1 0 1.5220 111.1000 0.0000 0.000
4 N N M 3 2 1 1.3350 116.6000 180.0000 -0.5200
5 HN H E 4 3 2 1.0100 119.8000 0.0000 0.2480
6 CA CH M 4 3 2 1.4490 121.9000 180.0000 0.2140
7 CB C2 S 6 4 3 1.5250 111.1000 60.0000 0.0380
8 CG CA S 7 6 4 1.5100 115.0000 180.0000 0.0110
9 CD1 CD S 8 7 6 1.4000 120.0000 180.0000 -0.0110
10 CE1 CD S 9 8 7 1.4000 120.0000 180.0000 0.0040
11 CZ CD S 10 9 8 1.4000 120.0000 0.0000 -0.0030
12 CE2 CD S 11 10 9 1.4000 120.0000 0.0000 0.0040
13 CD2 CD E 12 11 10 1.4000 120.0000 0.0000 -0.0110
14 C C M 6 4 3 1.5220 111.1000 180.0000 0.5260
15 O O E 14 6 4 1.2290 120.5000 0.0000 -0.5000

IMPROPER
-M CA N HN
CA +M C O
CB CA N C

LOOP
CG CD2

DONE
STOP

```

Figura A1.3.1.1.1 Imatge exemple de fitxer prepi del residu/monòmer PHE

AMBER²: Es pot trobar informació del programa de proteïnes a la següent url: <http://ambermd.org>

- Columna 1 – Comptador de les posicions del fitxer prepi.
- Columna 2 – Un nom atòmic exclusiu de l'àtom per a cada residu o monòmer.
- Columna 3 – Símbol del tipus d'àtom que defineix el seu comportament o interaccions atòmiques amb la resta d'àtoms del sistema. S'utilitza per assignar els paràmetres del camp de força (force-field). No s'utilitza en el programa SuSi.
- Columna 4 – El tipus topològic (símbol d'arbre) per l'àtom I (M, S, B, E, o 3)

Els àtoms en un residu es classifiquen en cinc tipus topològics: "Main", "Side", "Branch", "3", "4", "5", "6" i "Final". Es denoten com M, S, B, 3 4 5 6 i E respectivament.

Els àtoms principals descriuen el "camí" principal a través del residu, començant per la connexió amb el residu anterior i acabant en la connexió amb el següent residu. El programa connectarà l'últim àtom principal d'un residu al primer àtom principal del següent residu de la molècula. Si només hi ha un residu en una molècula, els àtoms principals solen ser la cadena continua més llarga. Els àtoms principals (M) poden tenir 1, 2, 3 o 4 àtoms connectats a ells.

Qualsevol àtom que no sigui un àtom principal es descriu per un dels altres tipus topològics: "E", "S", "B", "3", "4", "5" o "6". Un àtom "E" només té una connexió amb altres àtoms, de manera que és un "extrem sense sortida" per a qualsevol branca de qualsevol altre tipus d'àtom. Un àtom "S" ha de tenir un total de dues connexions amb altres àtoms, un àtom "B" ha de tenir un total de tres connexions, i un àtom "3" en realitat té un total de quatre connexions; el mateix s'aplica a "4" "5" i "6".

Example 1:

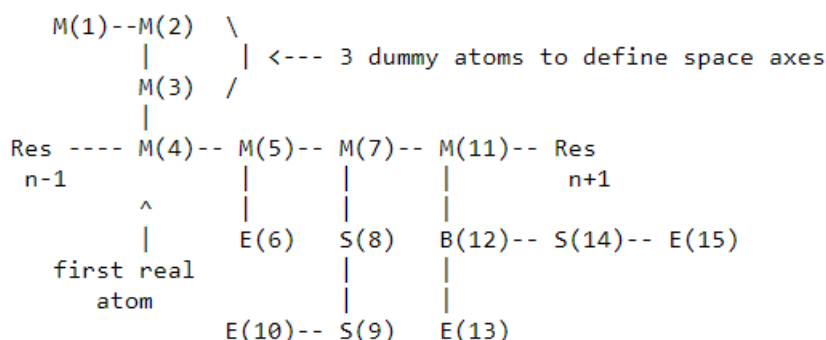


Figura A1.3.1.1.2 Imatge exemple de monòmer[<http://ambermd.org/doc/prep.html>]

- Columna 5 – NA(I) El número de l'àtom al què està connectat l'àtom "I".
- Columna 6 – NB(I) El número de l'àtom en què l'àtom "I" fa un angle juntament amb NA(I).
- Columna 7 – NC(I) El nombre de l'àtom en què l'àtom "I" fa un díedre juntament amb NA (I) i NB (I).
- Columna 8 – R(I) Aquesta és la longitud de connexió entre els àtoms "I" i NA (I).
- Columna 9 – THETA(I) és l'angle de connexió entre l'àtom NB (I), NA (I) i "I".
- Columna 10 – PHI(I) és l'angle diedre entre NC (I), NB (I), NA (I) i "I".
- Columna 11 – CHR(G) (I) La càrrega atòmica parcial de l'àtom "I".

Cal tenir en compte que els fitxers PREPI sempre esperen tres àtoms ficticis per començar.

Els àtoms ficticis precedeixin els àtoms reals del residu. Aquests àtoms només s'utilitzen per definir els eixos espacials del residu. Els tres àtoms ficticis han de tenir el tipus topològic "M", i se'ls ha d'assignar un tipus d'àtom de camp de força que els defineix com a àtoms ficticis.

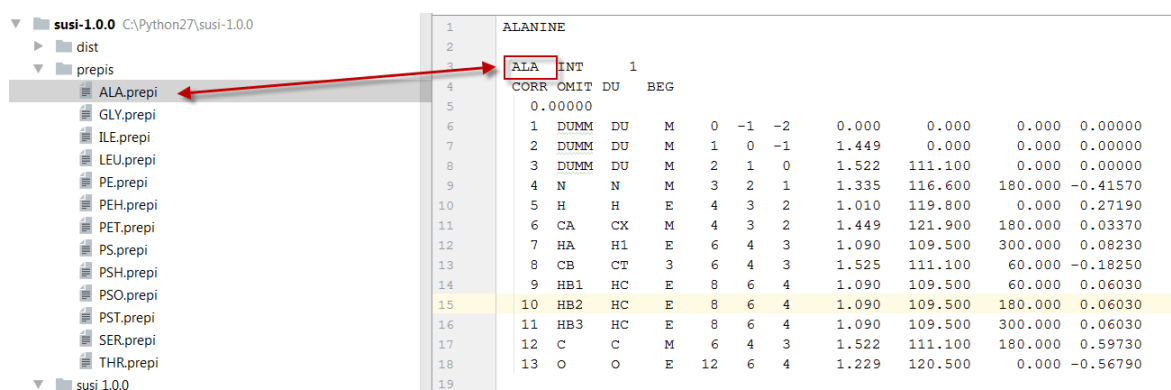
El programa desenvolupat utilitza el contingut de les columnes 5, 6, 7, 8, 9 i 10 per enllaçar els residus.

A1.3.1.3 Ubicació fitxers prepi

En la ubicació informada en el fitxer "path_prepi" cal desar tots els fitxers de residus o monòmers necessaris per la creació del sistema.

Per exemple si s'ha informat "path_prepi,prepi"; caldrà crear una carpeta amb aquest nom en el directori on es troben les classes de python del programa.

El nom dels fitxers prepi (ALA, GLY..) ha de coincidir amb el nom dels residus informats en el fitxer de càrrega amb les característiques del sistema per a que el programa els reconegui.



ALANINE										
1	2	3	4	5	6	7	8	9	10	11
ALA	INT	1								
	CORR	OMIT	DU	BEG						
0.00000										
1	DUMM	DU	M	0	-1	-2	0.000	0.000	0.000	0.00000
2	DUMM	DU	M	1	0	-1	1.449	0.000	0.000	0.00000
3	DUMM	DU	M	2	1	0	1.522	111.100	0.000	0.00000
4	N	N	M	3	2	1	1.335	116.600	180.000	-0.41570
5	H	H	E	4	3	2	1.010	119.800	0.000	0.27190
6	CA	CX	M	4	3	2	1.449	121.900	180.000	0.03370
7	HA	H1	E	6	4	3	1.090	109.500	300.000	0.08230
8	CB	CT	3	6	4	3	1.525	111.100	60.000	-0.18250
9	HB1	HC	E	8	6	4	1.090	109.500	60.000	0.06030
10	HB2	HC	E	8	6	4	1.090	109.500	180.000	0.06030
11	HB3	HC	E	8	6	4	1.090	109.500	300.000	0.06030
12	C	C	M	6	4	3	1.522	111.100	180.000	0.59730
13	O	O	E	12	6	4	1.229	120.500	0.000	-0.56790

Figura A1.3.1.2.1 Imatge exemple fitxer ALA.prepi

A1.3.1.4 Exemple d'un fitxer de càrrega

A continuació es mostra un fitxer de càrrega per a generar un sistema format per 4 molècules amb diferents residus (monòmers) d'aminoàcids, que anomenarem "exemple.txt":

```
system_name,10ALA
box,Lx 75 A,Ly 75 A,Lz 75 A
periodical_coordinates,Y
minimum_distance, 1.5 A
retries, 10
maximum_deviation, 5
```

```

molecules_number, 4
path_prepi, prepis
10 ALA, 5 GLY
10 ALA
10 ALA
10 ALA

```

Figura A1.3.1.4.1 Exemple fitxer càrrega “exemple.txt” amb les característiques del sistema

L'aplicació SuSi llegirà el fitxer input “exemple.txt” i començarà a processar les dades.

Amb el contingut d'aquest fitxer es crearà un PDB amb el nom “10ALA” (primera línia del fitxer, “system_name”) dins d'una caixa amb les mides $L_x=L_y=L_z$ de 75 Å (segona línia del fitxer, “box”), aplicant sempre les condicions límit periòdiques i respectant una distància mínima entre àtoms de 1,5 Å (quarta línia del fitxer, “minimum_distance”) per detectar col·lisió.

En cas de detectar col·lisió realitzarà 10 intents (cinquena línia del fitxer, “retries”) modificant posicions dels àtoms abans de descartar la possibilitat de col·locar el nou residu; i tindrà en compte una desviació màxima d'un 5% per realitzar els càlculs de les noves ubicacions.

Els fitxers amb els residus s'aniran a buscar a la carpeta “prepi” i es crearan 4 molècules amb la següent composició:

- A formada per 10 ALA i 5 GLY
- B formada per 10 ALA
- C formada per 10 ALA
- D formada per 10 ALA

A1.3.1.5 Exemple de fitxer de càrrega erroni.

Aquest fitxer no es processarà correctament sinó que donarà error per que manquen informar certs paràmetres:

```

system_name, 1ALAGLYALA
box, Lx 300 A, Ly 300 A, Lz 300 A
minimum_distance, 0.1 A
maximum_deviation, 5
molecules_number, 1
path_prepi, prepis
moll, ALA, GLY, ALA

```

Figura A1.3.1.5.1 Exemple fitxer erroni per la càrrega

En aquest cas el sistema donarà el següent missatge d'error:

“Errors found:File can't have less than 8 lines”

És imprescindible respectar l'ordre de les instruccions del fitxer de càrrega, ja que el programa sols és capaç de interpretar-lo si es respecta l'ordre correcte dels paràmetres.

Altres erros de lectura del fitxer es donaran quan no es trobin disponibles en el "path_prepi" tots els fitxers necessaris per la creació del sistema

"Errors found: Could not read file:prepis/10 ALA 7 PT.prepi"

En cas de no coincidir el número de molècules amb les posicions de molècules en el fitxer el procés també donarà un error:

"Errors found: Molecules number in file different than molecules number found"

Per últim cal tenir en compte que les distàncies sempre s'han d'informar en Àngstroms, en cas contrari el sistema també emetrà un error:

"Errors found: Error in Z dimension unit"

A1.3.2 Exemple d'execució del programa SuSi

Un cop es disposin dels fitxers prepi necessaris i aquests es trobin en el seu directori; s'hagi creat el fitxer de càrrega amb les característiques del sistema, es podrà procedir a la creació del fitxer PDB mitjançant la introducció de certes línies de comandes de Python.

A1.3.2.1 Pas 1: Accedir al directori de treball i iniciar el Python

Cal entrar a la carpeta on esta l'executable de Python i el fitxer Susi.py.

En el cas d'exemple el Python es va instal·lar dintre de la carpeta C:\Python27 d'un ordinador amb sistema operatiu Windows, per tant serà on estarà l'executable "python.exe" i a on també tindrem el fitxer Susi.py.

```
C:\Python27>
```

A1.3.2.2 Pas 2: Obtenir ajuda

Per obtenir l'ajuda que informa dels possibles paràmetres que es poden fer servir cal utilitzar el paràmetre -h (help).

Exemple: python Susi.py -h

```
C:\Python27>python Susi.py -h
usage: Susi.py [-h] [-t] [-v] -f FILE

optional arguments:
  -h, --help            show this help message and exit
  -t, --testmode        activate test mode, file.pdb won't be created
  -v, --verbosity        increase output verbosity
  -f FILE, --file FILE  input file with system properties
```

Figura A1.3.2.2.1 Instruccions per l'ajuda– Pas 2

A1.3.2.3 Pas 3a: Generació del sistema en mode real

Per generar el fitxer PDB del nou sistema cal executar aquesta comanda tot indicant el nom del fitxer de carrega.

Exemple: `python Susi.py -f create1ala.txt`

```
C:\Python27>python Susi.py -f create1ala.txt
Computing[.....] 0/10 Re
Computing[#####] 1/10 Re
Computing[#####] 2/10 Re
Computing[#####] 3/10 Re
Computing[#####] 4/10 Re
Computing[#####] 5/10 Re
Computing[#####] 6/10 Re
Computing[#####] 7/10 Re
Computing[#####] 8/10 Re
Computing[#####] 9/10 Re
Computing[#####] 10/10 R
sidues
### Process completed ###
No errors found
file 1ALA.pdb created
```

Figura A1.3.2.3.1 Instruccions mode real– Pas 3a

El missatge “file 1ALA.pdb created” ens indica que s’ha creat correctament el fitxer.

A1.3.2.4 Pas 3b: Generació del sistema en mode test

A vegades pot interessar realitzar només una simulació d’execució del programa per tal de testejar la correctesa del nostre fitxer de càrrega. Per realitzar el tractament en mode test cal fer servir el paràmetre -t (test) i també indicar el nom del fitxer de carrega amb el paràmetre -f (file).

Exemple: `python Susi.py -t -f create1ala.txt`


```

C:\Python27>python Susi.py -t -f create1ala.txt
No errors found
System name:      1ALA
X dimension size: 25.0 Å
Y dimension size: 25.0 Å
Z dimension size: 25.0 Å
Periodical coordinates: N
Minimum distance: 1.5 Å
Retries:          100
Maximum deviation: 5.0 %
Molecules number: 1
Path PREPI files: prepis
Molecules:        [['ALA']]
Residue list:      ['ALA']
prepi files ok:    ['prepis/ALA.prepi']

```

Figura A1.3.2.4.1 Instruccions mode test– Pas 3b

A1.3.2.5 Pas 3c: Generació del sistema en mode real amb logs

Per realitzar el tractament en mode normal però informant al usuari amb mes nivell de detall dels diferents passos realitzats pel programa SuSi, fem servir el paràmetre `-v` (Verbosity). Aquest paràmetre genera un fitxer “susi.log” amb els diferents detalls d’execució. També cal indicar el nom del fitxer de carrega amb el paràmetre `-f` (file).

Hi ha diferents nivells de “Verbosity” en funció del grau de detall en que es vulgui informar a l’usuari sobre la generació del nou sistema.

(nivell 0) : No hi ha missatges addicionals

`-v` (nivell 1): Petites indicacions de generació de les molècules del sistema

`-vv` (nivell 2): S’informa de tots els diferents intents bons/fallits per a la incorporació d’un nou residu.

Exemple: `python Susi.py -f create5ala.txt`

Exemple: `python Susi.py -v -f create5ala.txt`

```

C:\Python27>python Susi.py -t -f create1ala.txt
Computing[.....] 0/10 Res
Computing[#####] 1/10 Res
Computing[#####] 2/10 Res
Computing[#####] 3/10 Res
Computing[#####] 4/10 Res
Computing[#####] 5/10 Res
Computing[#####] 6/10 Res
Computing[#####] 7/10 Res
Computing[#####] 8/10 Res
Computing[#####] 9/10 Res
Computing[#####] 10/10 Res
Residues
### Process completed ###
No errors found
System name:          1ALA
X dimension size:     100.0 Å
Y dimension size:     50.0 Å
Z dimension size:     50.0 Å
Periodical coordinates: N
Minimum distance:     0.1 Å
Retries:              10
Maximum deviation:    5.0 %
Molecules number:     1
Path PREPI files:     prepis
Molecules:            [['ALA', 'ALA', 'ALA', 'ALA', 'ALA', 'ALA', 'ALA', 'ALA',
                        'ALA', 'ALA']]
Residue list:         ['ALA']
prepi files ok:       ['prepis/ALA.prepi']

```

Figura A1.3.2.5.1 Instruccions mode t– Pas 3c

Exemple: `python Susi.py -vv -f create5ala.txt`

El missatge “file 5ALA.pdb created” ens indica que s’ha creat correctament el fitxer.

A1.3.3 Resultats i exemples d’aplicació

A1.3.3.1 Procés mode real

Si el procés ha funcionat correctament, el programa retornarà un missatge indicant que s’ha creat un nou fitxer PDB. S’ha creat un PDB amb les característiques del sistema informat en el fitxer de càrrega.

En l’arrel del directori des d’on s’ha cridat el programa es crea un fitxer PDB amb el nom del sistema informat en el fitxer.

En funció del que s’ha informat en el paràmetres de coordenades periòdiques el programa dona com a resultat un fitxer PDB respectant les condicions de límit dins d’una caixa periòdica o sense tenir en

compte la caixa. En ambdós resultats s'ha realitzat el càlcul de les col·lisions dins de la caixa; i per tant tenint en compte les condicions de límit periòdiques.

A1.3.3.3 Exemple de fitxer PDB de sortida amb coordenades periòdiques

El sistema es troba dins de les dimensions de la caixa periòdica respectant les condicions de límit, tant per el PDB de sortida com per calcular les possibles col·lisions.

El contingut del fitxer de text on s'especifiquen les característiques del sistema és el següent; on s'ha ressaltat en negreta la variable per indicar que volem la sortida amb coordenades periòdiques:

```
system_name, PEH-60PE-PET2
box, Lx 75 A, Ly 50 A, Lz 50 A
periodical_coordinates, Y
minimum_distance, 0.1 A
retries, 10
maximum_deviation, 5
molecules_number, 6
path_prepi, prepis
PEH, 60 PE, PET
PEH, 60 PE, PET
PEH, 60 PE, PET
PEH, 60 PE, PET
PEH, 60 PE, PET
PEH, 60 PE, PET
```

Figura A1.3.3.2.1 Exemple fitxer amb coordenades periòdiques

En la carpeta “prepis” caldrà desar els residus PEH, PE i PET per a que s'executi correctament el programa. Les coordenades de generació seran diferents per a cada vegada que s'executi el programa SuSi. S'annexa una imatge de una part del PDB creat com a resultat i es mostra amb un visor poder veure el següent sistema dins de la caixa 3D:

1	ATOM	1	C1	FEH	A	0	0.860	2.740	-2.723	1.00	0.00	C
2	ATOM	2	H9	FEH	A	0	-0.118	2.735	-3.193	1.00	0.00	H
3	ATOM	3	H10	FEH	A	0	1.597	2.132	-3.241	1.00	0.00	H
4	ATOM	4	H11	FEH	A	0	0.734	2.369	-1.710	1.00	0.00	H
5	ATOM	5	C2	FEH	A	0	1.330	4.195	-2.722	1.00	0.00	C
6	ATOM	6	H12	FEH	A	0	2.311	4.250	-2.259	1.00	0.00	H
7	ATOM	7	H13	FEH	A	0	0.581	4.815	-2.234	1.00	0.00	H
8	ATOM	8	C3	FE	A	1	1.507	4.674	-4.163	1.00	0.00	C
9	ATOM	9	H14	FE	A	1	0.544	4.636	-4.668	1.00	0.00	H
10	ATOM	10	H15	FE	A	1	2.282	4.070	-4.633	1.00	0.00	H
11	ATOM	11	C4	FE	A	1	1.938	6.141	-4.163	1.00	0.00	C
12	ATOM	12	H16	FE	A	1	2.901	6.225	-3.664	1.00	0.00	H
13	ATOM	13	H17	FE	A	1	1.144	6.731	-3.707	1.00	0.00	H
14	ATOM	14	C3	FE	A	2	2.155	6.608	-5.603	1.00	0.00	C
15	ATOM	15	H14	FE	A	2	1.212	6.539	-6.141	1.00	0.00	H
16	ATOM	16	H15	FE	A	2	2.962	6.021	-6.036	1.00	0.00	H
17	ATOM	17	C4	FE	A	2	2.545	8.087	-5.605	1.00	0.00	C
18	ATOM	18	H16	FE	A	2	3.487	8.203	-5.072	1.00	0.00	H
19	ATOM	19	H17	FE	A	2	1.720	8.660	-5.185	1.00	0.00	H
20	ATOM	20	C3	FE	A	3	2.802	8.544	-7.041	1.00	0.00	C
21	ATOM	21	H14	FE	A	3	1.882	8.442	-7.613	1.00	0.00	H
22	ATOM	22	H15	FE	A	3	3.640	7.973	-7.438	1.00	0.00	H
23	ATOM	23	C4	FE	A	3	3.152	10.032	-7.047	1.00	0.00	C
24	ATOM	24	H16	FE	A	3	4.070	10.180	-6.482	1.00	0.00	H
25	ATOM	25	H17	FE	A	3	2.297	10.588	-6.665	1.00	0.00	H
26	ATOM	26	C3	FE	A	4	3.449	10.480	-8.478	1.00	0.00	C
27	ATOM	27	H14	FE	A	4	2.554	10.347	-9.082	1.00	0.00	H
28	ATOM	28	H15	FE	A	4	4.317	9.928	-8.838	1.00	0.00	H
29	ATOM	29	C4	FE	A	4	3.759	11.977	-8.490	1.00	0.00	C
30	ATOM	30	H16	FE	A	4	4.652	12.156	-7.894	1.00	0.00	H
31	ATOM	31	H17	FE	A	4	2.876	12.513	-8.146	1.00	0.00	H
32	ATOM	32	C3	FE	A	5	4.097	12.416	-9.915	1.00	0.00	C
33	ATOM	33	H14	FE	A	5	3.228	12.252	-10.549	1.00	0.00	H
34	ATOM	34	H15	FE	A	5	4.992	11.885	-10.235	1.00	0.00	H
35	ATOM	35	C4	FE	A	5	4.366	13.921	-9.933	1.00	0.00	C
36	ATOM	36	H16	FE	A	5	5.231	14.132	-9.308	1.00	0.00	H

Figura A1.3.3.2.2 Exemple fitxer PDB

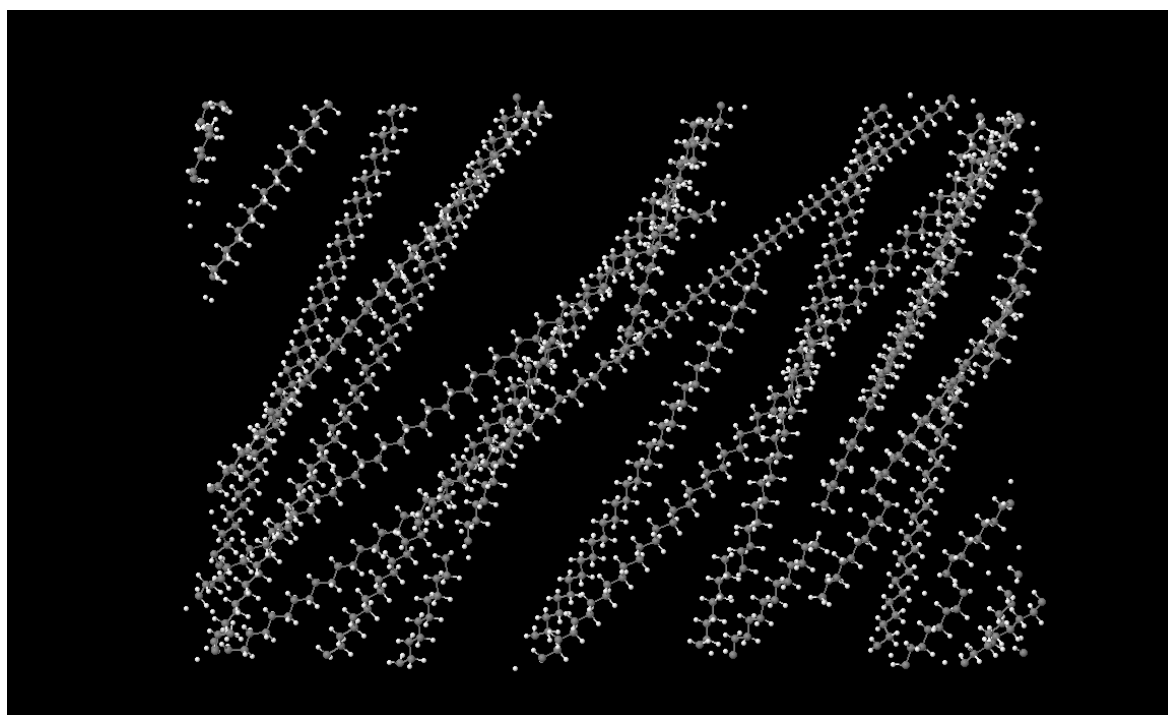


Figura A1.3.3.2.3 Representació del sistema dins de la caixa periòdica

A1.3.3.3 Fitxer sense coordenades periòdiques

A vegades pot interessar de generar les coordenades d'un sistema periòdic però que les partícules no estiguin encabides dins de les caixa periòdica, tot i que s'ha tingut en compte les condicions de periodicitat per calcular les possibles col·lisions. És a dir, les partícules no estan circumscrites dins de la cel·la unitària i no apareixen enllaços/molècules trencades.

El contingut del fitxer de text on s'especifiquen les característiques del sistema és el següent; on s'ha ressaltat en negreta la variable per indicar que no volem la sortida amb coordenades periòdiques:

```
system_name, PEH-60PE-PET2
box, Lx 75 A, Ly 50 A, Lz 50 A
periodical_coordinates, N
minimum_distance, 0.1 A
retries, 10
maximum_deviation, 5
molecules_number, 6
path_prepi, prepi
PEH, 60 PE, PET
PEH, 60 PE, PET
PEH, 60 PE, PET
PEH, 60 PE, PET
PEH, 60 PE, PET
PEH, 60 PE, PET
```

Figura A1.3.3.3.1 Exemple fitxer amb coordenades NO periòdiques

En la carpeta “prepi” caldrà desar els residus PEH, PE, i PET per a que s'executi correctament el programa. S'adjunta el fitxer PDB creat segons el següent fitxer de càrrega:

1	ATOM	1	C1	PEH	A	0	-1.564	2.822	1.724	1.00	0.00	C
2	ATOM	2	H9	PEH	A	0	-1.568	2.473	2.751	1.00	0.00	H
3	ATOM	3	H10	PEH	A	0	-2.547	2.821	1.259	1.00	0.00	H
4	ATOM	4	H11	PEH	A	0	-0.899	2.171	1.164	1.00	0.00	H
5	ATOM	5	C2	PEH	A	0	-1.033	4.255	1.758	1.00	0.00	C
6	ATOM	6	H12	PEH	A	0	-1.012	4.649	0.746	1.00	0.00	H
7	ATOM	7	H13	PEH	A	0	-0.064	4.268	2.253	1.00	0.00	H
8	ATOM	8	C3	PE	A	1	-2.002	5.139	2.543	1.00	0.00	C
9	ATOM	9	H14	PE	A	1	-2.054	4.780	3.569	1.00	0.00	H
10	ATOM	10	H15	PE	A	1	-2.958	5.147	2.023	1.00	0.00	H
11	ATOM	11	C4	PE	A	1	-1.450	6.563	2.613	1.00	0.00	C
12	ATOM	12	H16	PE	A	1	-1.380	6.963	1.603	1.00	0.00	H
13	ATOM	13	H17	PE	A	1	-0.504	6.539	3.152	1.00	0.00	H
14	ATOM	14	C3	PE	A	2	-2.439	7.457	3.363	1.00	0.00	C
15	ATOM	15	H14	PE	A	2	-2.542	7.091	4.382	1.00	0.00	H
16	ATOM	16	H15	PE	A	2	-3.371	7.484	2.801	1.00	0.00	H
17	ATOM	17	C4	PE	A	2	-1.868	8.872	3.468	1.00	0.00	C
18	ATOM	18	H16	PE	A	2	-1.748	9.278	2.466	1.00	0.00	H
19	ATOM	19	H17	PE	A	2	-0.947	8.828	4.047	1.00	0.00	H
20	ATOM	20	C3	PE	A	3	-2.875	9.775	4.181	1.00	0.00	C
21	ATOM	21	H14	PE	A	3	-3.028	9.403	5.192	1.00	0.00	H
22	ATOM	22	H15	PE	A	3	-3.781	9.821	3.580	1.00	0.00	H
23	ATOM	23	C4	PE	A	3	-2.287	11.179	4.323	1.00	0.00	C
24	ATOM	24	H16	PE	A	3	-2.116	11.592	3.331	1.00	0.00	H
25	ATOM	25	H17	PE	A	3	-1.393	11.117	4.942	1.00	0.00	H
26	ATOM	26	C3	PE	A	4	-3.310	12.093	5.000	1.00	0.00	C
27	ATOM	27	H14	PE	A	4	-3.512	11.716	6.000	1.00	0.00	H
28	ATOM	28	H15	PE	A	4	-4.188	12.159	4.360	1.00	0.00	H
29	ATOM	29	C4	PE	A	4	-2.707	13.487	5.178	1.00	0.00	C
30	ATOM	30	H16	PE	A	4	-2.486	13.904	4.198	1.00	0.00	H
31	ATOM	31	H17	PE	A	4	-1.842	13.406	5.835	1.00	0.00	H
32	ATOM	32	C3	PE	A	5	-3.743	14.412	5.818	1.00	0.00	C
33	ATOM	33	H14	PE	A	5	-3.995	14.030	6.805	1.00	0.00	H

Figura A1.3.3.3.2 Exemple fitxer PDB amb coordenades NO periòdiques

Finalment es mostra el PDB amb un visor on es veu que el sistema no es troba dins de la caixa 3D:

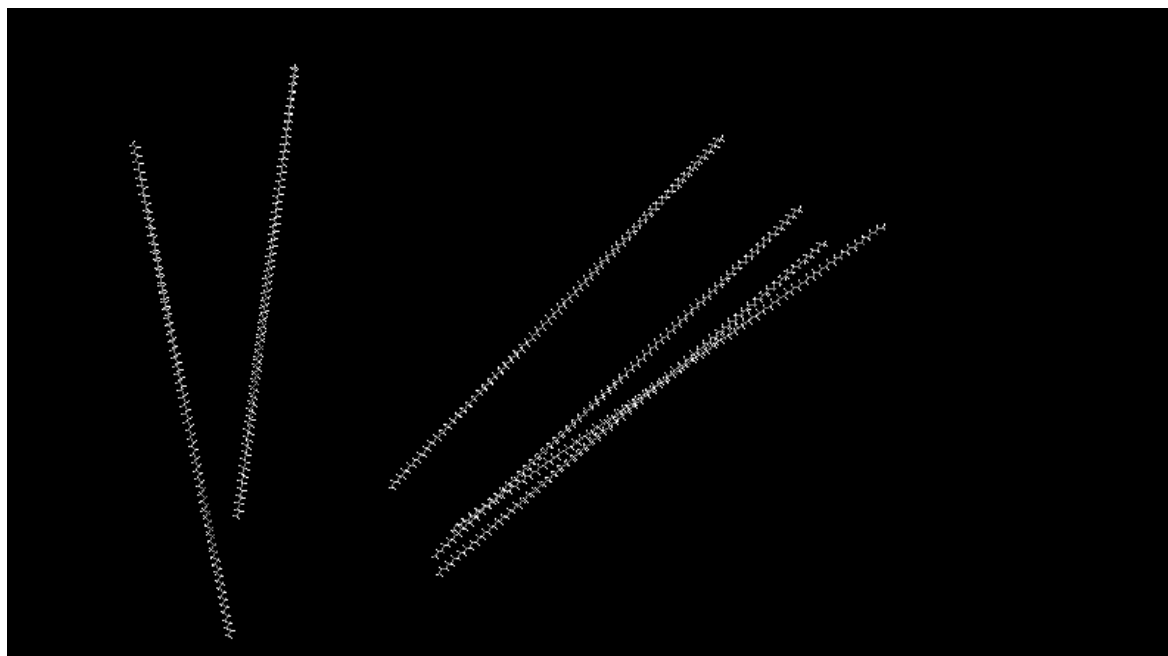


Figura A1.3.3.3.3 Representació del sistema sense tenir en compte els límits de la caixa

A1.3.3.4 Execució en mode test

L'execució en mode test serveix per comprovar que el fitxer de càrrega està escrit correctament, mai crearà el fitxer PDB. Si detecta algun error el mostrarà per pantalla però si tot funciona correctament mostrarà per pantalla informació relacionada amb el programa de carrega. Exemple sortida :

```
No errors found
System name:      2ILE
X dimension size: 100.0 A
Y dimension size: 100.0 A
Z dimension size: 100.0 A
Periodical coordinates: Y
Minimum distance: 15.0 A
Retries:          100
Maximum deviation: 5.0 %
Molecules number: 1
Path PREPI files: prepi
Molecules:        [['ILE', 'ILE']]
Residue list:      ['ILE']
prepi files ok:    ['prepi/ILE.prepi']

Process finished with exit code 0
```

Figura A1.3.3.4.1 Exemple sortida mode test

Explicació de la informació que es mostra:

Resultat	Valor
No errors found	
System name:	Mostra nom del sistema informat en el fitxer
X dimension size:	Mostra la unitat Lx informada en el fitxer
Y dimension size:	Mostra la unitat Ly informada en el fitxer
Z dimension size:	Mostra la unitat Lz informada en el fitxer
Periodical coordinates:	Mostra el valor de la variable de coordenades periòdiques informada en el fitxer
Minimum distance:	Mostra la mínima distància informada en el fitxer
Retries:	Monstre el número de reintents informats en el fitxer
Maximum deviation:	Mostra el % de desviació màxima informada en el fitxer
Molecules number:	Mostra el número de molècules informades en el fitxer
Path PREPI files:	Mostra les diferents rutes que cal anar a llegir per obtenir els fitxers prepi
Molecules:	Mostra la composició de cada una de les molècules
Residue list:	Mostra la llista de residus que cal anar a llegir a la carpeta
prepi files ok:	Indica que s'ha trobat el directori on es troben els fitxers prepi.

Figura A1.3.3.4.2 Taula amb els valors que mostra el mode test

A1.3.4 Llista de possibles errors

Si el procés no s'executa correctament, no es crea el fitxer PDB sinó que mostra l'error que ha fet aturar la generació de les coordenades finals del sistema.

Els errors estan numerats i es troben enumerats en la classe message en anglès; però es podrien fàcilment configurar en altres idiomes. A continuació es mostra una llista amb els errors que detecta el programa i una taula amb la seva possible solució:

- '001' : 'File can't have less than 8 lines',
- '002' : 'Collisions could not be avoided for this molecule',
- '003' : 'system_name not found in file',
- '004' : 'box dimension not found',
- '005' : 'periodical_coordinates not found',
- '006' : 'minimum_distance not found',
- '007' : 'retries not found in file',
- '008' : 'maximum_deviation not found in file',
- '009' : 'molecules_number not found in file',
- '010' : 'PATH of prepri files not found in file',
- '011' : 'Error in X dimension unit',
- '012' : 'Error in Y dimension unit',
- '013' : '',
- '014' : 'Error in Z dimension unit',
- '015' : 'Error in minimum distance unit',
- '016' : 'System name not found in file',
- '017' : 'Error in value of periodical coordinates. Possible values are Y or N',
- '018' : 'Error in minimum distance'
- '019' : 'Molecules number not found in file',
- '020' : 'PATH of prepri files not found in file',
- '021' : 'Molecules number in file different than molecules number found',
- '022' : 'Error in box',
- '023' : 'Could not read file: &',
- '024' : 'No errors found',
- '025' : '',
- '026' : '',
- '027' : ''

A1.3.4.1 Errors detectats en la lectura del fitxer de càrrega

Num	Error	Solució error
001	File can't have less than 8 lines	El fitxer ha de tenir informats tots els paràmetres necessaris, i per tant com a mínim ha de tenir 8 línies.
003	System name not found in file.	No s'ha indicat el nom del sistema en el fitxer de càrrega.
004	Box dimension not found.	No s'han informat les dimensions de la

		caixa en el fitxer de càrrega.
005	Periodical coordinates not found.	No s'han informat les coordenades periòdiques en el fitxer de càrrega.
006	Minimum distance not found.	No s'ha informat la distància mínima de col·lisió en el de càrrega.
007	retries not found in file	No s'ha informat el número de re-intents en el fitxer de càrrega.
008	Maximum deviation not found in file.	No s'ha informat la desviació màxima en el fitxer de càrrega.
009	Molecules number not found in file.	No s'ha informat el número de molècules en el fitxer de càrrega.
010	PATH of prepri files not found in file.	El directori on estan els fitxers prepis no s'ha trobat
011	Error in X dimension unit.	La unitat de mesura en la coordenada X de la caixa no és correcte, ja que han de ser àngstroms
012	Error in Y dimension unit.	La unitat de mesura en la coordenada Y de la caixa no és correcte, ja que han de ser àngstroms
014	Error in Z dimension unit.	La unitat de mesura en la coordenada Z de la caixa no és correcte, ja que han de ser àngstroms
015	Error in minimum distance unit.	La unitat de mesura de la distància mínima no és correcte, ja que han de ser àngstroms
016	System name not found in file	No s'ha informat el nom del sistema en el fitxer.
017	Error in value of periodical coordinates. Possible values are Y or N	S'ha informat un valor erroni en el paràmetre de coordenades periòdiques. Sols pot ser S o N.
018	Error in minimum distance.	No s'ha informat correctament la distància mínima, s'ha d'informar la quantitat i la unitat.
019	Molecules number not found in file	No s'ha informat el número de de molècules en el fitxer.
020	'PATH of prepri files not found in file',	No s'ha trobat el directori on estan emmagatzemats els prepis.
021	Molecules number in file different than molecules number found.	El número de molècules informades no és el mateix que l'indicat en el fitxer en la variable "molecules_number".
022	Error in box	Hi ha algun problema amb els paràmetres de la caixa. Revisa mesures i les seves unitats.

Figura A1.3.4.1.1 Taula errors fitxer de càrrega

A1.3.4.2 Altres errors detectats en l'execució.

Num	Error	Solució error
001	Collisions could not be avoided for this molecule	Quan no és possible evitar de cap manera les col·lisions per crear la molècula
	File xxxxxx.pdb exists	El fitxer de carrega és correcte, però en intentar generar el fitxer PDB s'ha trobat que ja existia un amb el mateix nom.
023	Could not read file: &'	No es pot llegir el fitxer especificat
024	No errors found	No s'ha trobat cap error al executar el programa

Figura A1.3.4.1.2 Taula error

Annex A2 – Detall tècnic: Atributs i funcions

En aquest annex es detallaran breument els diferents conceptes que formen el conjunt dels elements desenvolupats en el projecte; així com els atributs de cada classe i una descripció breu de la funcionalitat de les seves funcions.

S'ha dividit el document en 3 seccions:

- Classes principals.
- Classes d'entrada i sortida.
- Classes de processament.

A2.1 Classes principals

Les classes principals són les que serveixen per definir quins elements tindrà un sistema. La relació entre les diferents classes ens indica el número d'elements que podrà tenir el sistema.

Un sistema podrà tenir una o més molècules, una molècula podrà estar formada per un o més residus i un residu podrà estar format per un o més àtoms.

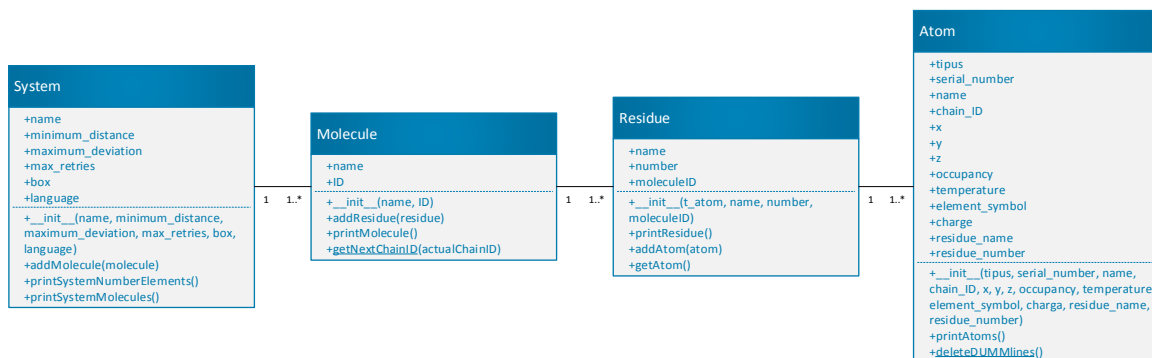


Diagrama A2.1.1. Model conceptual classes principals

A continuació es detallaran breument els atributs i funcions que tenen cada una de les classes que formen el domini del programa.

A2.1.1 SYSTEM

La classe SYSTEM és la classe més important de les classes principals. A partir d'aquesta classe podem relacionar la resta d'elements que formen un sistema: Molècules, residus i àtoms.

Els atributs de la classe són:

Atribut	Descripció
t_molecules	Llista amb les molècules que tindrà el sistema. Un sistema podrà contenir una o més molècules.
name	Nom del sistema.
minimum_distance	Distància mínima a la que es considerarà que s'ha produït una col·lisió.
maximum_deviation	Desviació mínima.
max_retries	Número d'intents màxims que s'intentarà situar un residu per evitar la col·lisió.
box	Dimensions i unitats de la caixa on esta la molècula. Box[0]: Llargada posició X. Box[1]: Unitat de la X. Actualment han de ser Armstrong. Box[2]: Llargada posició Y. Box[3]: Unitat de la Y. Actualment han de ser Armstrong. Box[4]: Llargada posició Z. Box[5]: Unitat de la Z. Actualment han de ser Armstrong.
language	Idioma pels missatges. Exemple 'EN' = Anglès.

Taula A2.1.1.1 Atributs de la classe SYSTEM.

Les funcions de la classe son:

Funció	Descripció
Init	Permet declarar una variable de tipus sistema.
addMolecule	Afegeix una molècula a la llista de molècules de la classe (t_molecules).
printSystemNumberElements	Mostra per pantalla informació relacionada amb el sistema.
printSystemMolecules	Mostra per pantalla el contingut del sistema. És molt útil a l'hora de comprovar si el tractament del sistema s'està fent correctament

Taula A2.1.1.2. Funcions de la classe SYSTEM.

A2.1.2 MOLECULE

Un sistema podrà estar format per una o mes molècules. Una molècula la podrem identificar pel seu ID de molècula que estarà format per una lletra del abecedari, començant per la A. Es a dir, la primera molècula d'un sistema serà la A, la següent molècula tindrà el ID de molècula B, i així successivament. La determinació d'aquest ID de molècula s'aconsegueix mitjançant una de les funcions d'aquesta classe, la funció getNextChain.

Els atributs de la classe son:

Atribut	Descripció
t_residues	Llista amb els residus que conté la molècula, és a dir, una molècula podrà contenir un o més residus.
name	Nom de la molècula.
ID	ID de la molècula. Aquest ID està format per una lletra de l'abecedari.

Taula A2.1.2.1. Atributs de la classe Molecule

Les funcions de la classe son:

Funció	Descripció
init	Permet declarar una variable de tipus molècula.
addResidue	Afegeix un residu a la llista de residus de la molècula (t_residues).
printMolecule	Mostra per pantalla tota la informació relativa a la molècula.
getNextChain	Obté el següent valor del ID de la molècula; per exemple, si el valor actual és una D, el següent valor serà una E.

Taula A2.1.2.2 Funcions de la classe Molecule

A2.1.3 RESIDUE

Un residu pertany a una molècula del sistema, i podrà estar format per un o més àtoms.

Els atributs de la classe son:

Atribut	Descripció
t_atom	Llista amb els àtoms que conté el residu. Un residu podrà contenir un o més àtoms.
name	Nom del residu.
number	Número del residu dins de la molècula.
moleculeID	ID de la molècula a la que pertany el residu.

Taula A2.1.3.1. Atributs de la classe RESIDUE.

Les funcions de la classe son:

Funció	Descripció
init	Permet declarar una variable de tipus residu.
printResidue	Mostra per pantalla tota la informació relativa al residu.
addAtom	Afegeix un registre a la llista d'àtoms (t_atom).
getAtom	Obté les dades d'un àtom determinat.

Taula A2.1.3.2. Funcions de la classe RESIDUE.

A2.1.4 ATOM

Un àtom pertany a un únic residu i estarà format per una sèrie d'atributs que posteriorment seran necessaris per la creació de les línies que formaran el fitxer PDB.

Alguns d'aquests atributs que són més aviat referents a un PDB, i no tant a la informació referent al propi àtom en qüestió, com ara Occupancy i temperatura.

Els atributs de la classe ATOM son els següents:

Atribut	Descripció
tipus	Valor del tipus d'àtom.
serial_number	Nombre de sèrie.

name	Nom de l'àtom.
chain_ID	Valor del chain ID.
x	Posició X on està situat l'àtom.
y	Posició y on està situat l'àtom.
z	Posició z on està situat l'àtom.
occupancy	Occupancy.
temperature	Valor de la temperatura.
element_symbol	Símbol.
charge	Càrrega.
residue_name	Nom del residu al que pertany l'àtom.
residue_number	Número del residu al que pertany l'àtom.

Taula A2.1.4.1. Atributs de la classe ATOM.

Les funcions de la classe son:

Funció	Descripció
init	Permet declarar una variable de tipus àtom.
printAtom	Mostra per pantalla el contingut d'un àtom.
deteleDUMMlines	Esborra els registres DUMM que estan dins del array_t_atom.

Taula A2.1.4.2. Funcions de la classe ATOM.

A2.2 Classes d'entrada i sortida

En aquest apartat es detallen les classes del sistema que s'utilitzen per realitzar el Input / Output; és a dir per a la lectura de dades a processar i l'escriptura dels resultats.

Com es mostra en el diagrama des de la classe principal "FILES" es criden les subclasses "FILEPDB" o "FILEPREPI" en funció del tipus de fitxer que s'està tractant, ja sigui per lectura o escriptura.

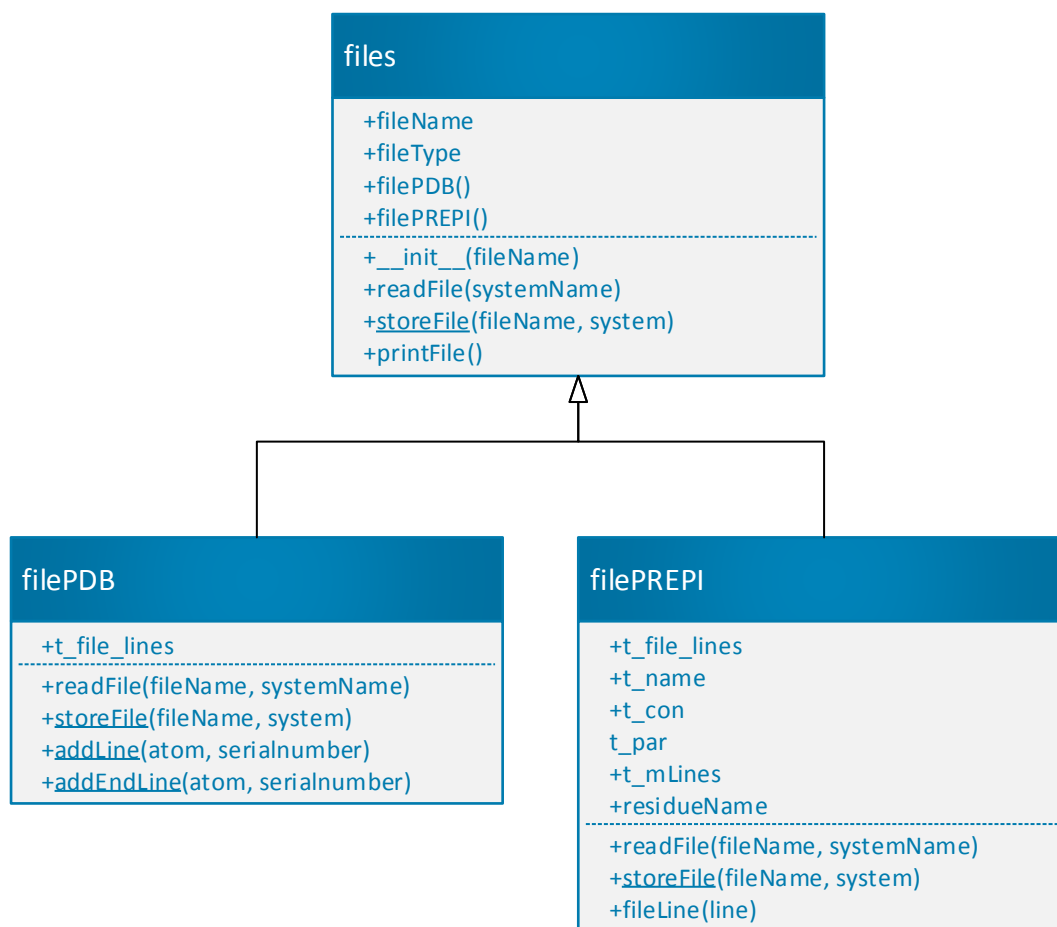


Diagrama A2.2.1. Model conceptual classes input /output

A continuació es detallaran breument els atributs i funcions que tenen cada una de les classes que formen el input/output del programa.

A2.2.1 FILES

La classe Files es fa servir pel tractament de fitxers. Aquesta classe crida a les classes FILEPDB i FILEPREPI.

En aquesta classe s'engloba tot el relacionat amb el tractaments de fitxers d'una forma genèrica. Els tipus de fitxers que es poden tractar actualment són PDB i PREPI; i en funció del tipus de fitxer s'utilitzaran unes funcions o altres; i també en funció del tipus de fitxer es cridarà a la classe FILEPDB o a la classe FILEPREPI.

Els atributs de la classe son:

Atribut	Descripció
fileName	Nom del fitxer.
fileType	Tipus de fitxer. Podrà ser fitxer PDB o PREPI.
filePDB	Variable tipus classe FILEPDB.
filePREPI	Variable tipus classe FILEPREPI.

Taula A2.2.1.1. Atributs de la classe FILES.

Les funcions de la classe son:

Funció	Descripció
Init	Detecta el tipus de fitxer i fa la crida al mètode de la classe del fitxer.
readFile	Detecta el tipus de fitxer i fa la crida al mètode de la classe del fitxer.
storeFile	Detecta el tipus de fitxer i fa la crida al mètode de la classe del fitxer.
printFile	Imprimeix el contingut del fitxer per pantalla.

Taula A2.2.1.2. Funcions de la classe FILES.

A2.2.2 FILEPDB

Aquesta classe inclou les funcions pel tractament dels fitxers PDB. La crida a aquesta classe es fa sempre des de la classe Files.

Els atributs de la classe son:

Atribut	Descripció
t_file_lines	Llista amb les línies que formen el contingut del fitxer.

Taula A2.2.2.1. Atributs de la classe FILEPDB.

Les funcions de la classe son:

Funció	Descripció
readFile	Llegeix el contingut del fitxer.

storeFile	Emmagatzema el fitxer.
addLine	Retorna en una línia el contingut de l'àtom. Es fa servir en el moment de la creació del PDB.
addEndLine	Retorna en una línia part del contingut que forma l'àtom. També es fa servir en el moment de la creació del fitxer PDB.

Taula A2.2.2.2. Funcions de la classe FILEPDB.

A2.2.3 FILEPREPI

Aquesta classe inclou les funcions pel tractament dels fitxers Prepi. La crida a aquesta classe es fa sempre des de la classe Files.

Cada cop que es llegeix un fitxer PREPI, s'ompliran els atributs de la classe.

Els atributs de la classe son:

Atribut	Descripció
name	Nom del fitxer.
t_file_lines	Llista amb les línies que formen el contingut del fitxer.
t_name	Llista amb els noms dels àtoms que conté el fitxer.
t_con	Llista amb els valors na, nb i nc de les línies del PREPI. Cada element de la llista conté 3 valors: <ul style="list-style-type: none"> La posició 1 correspon al valor de la columna na del fitxer PREPI. La posició 2 correspon al valor de la columna nb del fitxer PREPI. La posició 3 correspon al valor de la columna nc del fitxer PREPI.
t_par	Llista amb els valors r, theta i phi de les línies del PREPI. Cada element de la llista conté 3 valors: <ul style="list-style-type: none"> La posició 1 correspon al valor de la columna r del fitxer PREPI. La posició 2 correspon al valor de la columna theta del fitxer PREPI. La posició 3 correspon al valor de la columna phi del fitxer PREPI.
t_mLines	Llista amb el número de les línies que tenen el valor M a la quarta columna del fitxer PREPI.

Taula A2.2.3.1. Atributs de la classe FILEPREPI.

Les funcions de la classe son:

Funció	Descripció
Init	Inicialitza la classe del fitxer.
readFile	Llegeix el contingut del fitxer.
trataFile	Tracta el contingut del fitxer.
getCar	Per obtenir les darreres línies M.
getPar	Per obtenir les darreres línies M.
checkFile	Imprimeix per pantalla els atributs del fitxer.
StoreFile	Espai preparat per si es vol implementar la funció.

Taula A2.2.3.2. Funcions de la classe FILEPREPI.

A2.2.4 MESSAGES

Aquesta classe permetrà el tractament dels missatges en diferents idiomes. Actualment s'han creat tots els missatges en anglès, però si posteriorment es volen afegir altres llenguatges es podrà fer modificant la funció `getTextMessage` inclosa dintre d'aquesta classe.



Diagrama A2.2.4.1 Model conceptual classe de MESSAGES

Les funcions de la classe son:

Funció	Descripció
getTextMessage	Obté el text del missatge en el llenguatge sol·licitat.

Taula A2.2.4.1. Funcions de la classe FILEPREPI

A2.3 Classes de processament

Les classes de processament permeten relacionar totes les classes del projecte.

A continuació es detallaran breument els atributs i funcions que tenen cada una de les classes que formen el input/output del programa.

A2.3.1 __INIT__

Aquest fitxer serveix per importar les classes, fent-les visibles de tal manera que puguin ser utilitzades per les altres classes. Per a que funcioni correctament cal incloure una línia per cada una de les classes que es van creant i que formen part de l'aplicació.

A2.3.2 INT2CAR_M

Aquesta classe transforma les coordenades internes dels àtoms del fitxer PREPI en coordenades cartesianes (x,y,z)

Les funcions de la classe son:

Funció	Descripció
int2carX	Obté el valor de la posició X.
int2cary	Obté el valor de la posició Y.
int2carz	Obté el valor de la posició Z.
new_car	Obté el valor de les coordenades X, Y, Z.
print_xyz	Mostrà per pantalla el contingut de les variables.
int2car	Obté el valor actual per XYZ.

Taula A2.3.2.1. Funcions de la classe INT2CAR.

A2.3.3 BUILDER

La classe BUILDER es l'encarregada de llegir el fitxer d'entrada, els fitxers PREPI i de realitzar la crida a la resta de funcions d'altres classes per crear el fitxer PDB.

builder

```
+system
+system2
+verbosity
+logFile
-----
+__init__(fileName, verb, test)
+getAndCheckFileValues(t_lines)
+checkRepeatedResidues(t_residues)
-__addNewMolecule(t_molecule, moleculeNumber, moleculeID, t_filePREPI, t_log)
-__processResidue(numResidue, residueName, numRetry, moleculeNumber, t_filePREPI, t_xyz, t_last3m, t_log)
-__addNewResidue(t_xyz, filePREPI, numResidue, moleculeID, t_log)
+convertxyz(t_xyz)
+convertxyz2(t_xyz)
+convertToAtoms(t_xyz, t_atomName, residueName, residueNumber, chainID)
+printErrors(t_errors, language)
+getLast3M(nLines, t_xyz, numResidue, t_oldLast3m)
+PeriodicBoundaryCondition(box, posX, posY, posZ)
-__checkCollisionInSystem(moleculeID, atom, t_log)
-__checkCollisionInSystem2(posX, posY, posZ)
-__getRandomPositionWithoutCollision()
+getRandomThetaAndPhi(t_par, maximumDeviation)
+getSystem()
```

Diagrama A2.3.3.1. Model conceptual classe de MESSAGES

Els atributs de la classe son:

Atribut	Descripció
system	Variable de tipus classe SYSTEM.
system2	Variable de tipus classe SYSTEM.
verbosity	Indicarà si volem visualitzar el detall del procés.
logFile	Llista amb el detall del procés que ha realitzat la classe.

Taula A2.3.3.2. Atributs de la classe BUILDER.

Les funcions de la classe son:

Funció	Descripció
__init__	Realitza el tractament principal de la classe. S'explica en detall més endavant.
getAndCheckFileValues	Comprova la informació del fitxer d'entrada i retorna errors en cas de que no sigui correcta.
checkRepeatedResidues	Realitza el tractament dels residus indicats en les línies del fitxer d'entrada.

__addNewMolecule	Inicia el tractament per afegir una nova molècula al sistema que s'està creant.
__processResidue	Continua el tractament d'afegir una nova molècula iniciat per __addNewMolecule.
__addNewResidue	Afegeix un nou residu al sistema que s'està creant sempre i quan no hi hagi col·lisió entre els àtoms del sistema.
convertxyz + convertxyz2	La classe int2car retorna les coordenades en un format determinat, però necessitem convertir aquests valors en un format de números de 8 posicions amb 3 decimals.
convertToAtoms	Transforma el contingut obtingut per la funció convertxyz al format que posteriorment es farà servir per la creació del fitxer PDB.
printErrors	Mostra per pantalla els errors que s'han detectat
getLast3M	Cerca les coordenades cartesianes dels darrers àtoms del fitxer PREPI que s'està processant.
periodicBoundaryCondition	Calcula si la posició xyz es troba dins del box, i en cas contrari es recalcula la posició mirall
__checkCollisionInSystem	Comprova si la posició del àtom col·lidiona amb algun altre àtom que es trobi dins del cub.
__checkCollisionInSystem2	Comprova si la posició d'un àtom donat col·lidiona amb algun altre àtom que es trobi dins del sistema (polímer en construcció)
__getRandomPositionWithoutCollision	Obté una posició aleatòria que no col·lidiona amb cap àtom del sistema
getRandomThetaAndPhi	Serveix per obtenir valors aleatoris gracies a la funció randomValue.
getSystem	Retorna una variable de tipus System per poder tractar els seus atributs.

Taula A2.3.3.2 Funcions de la classe BUILDER.

Annex A3 – Versió de distribució

En aquest annex s'explica breument com crear una versió de distribució de Python:

En aquest annex es detallaran breument la forma de crear una versió de distribució de les classes de Python.

La versió de distribució serveix per instal·lar les classes del projecte en altres ordinadors. Els passos a realitzar són el següents:

A3.1 Setup.py

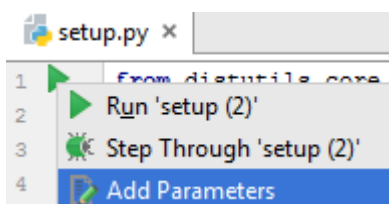
Crear un fitxer setup.py en l'arrel dels fitxers Python

```
from distutils.core import setup
setup(name='susi',
      version='1.0.0',
      description='Aplicacio matrius polimeriques',
      packages=['susi']
    )
```

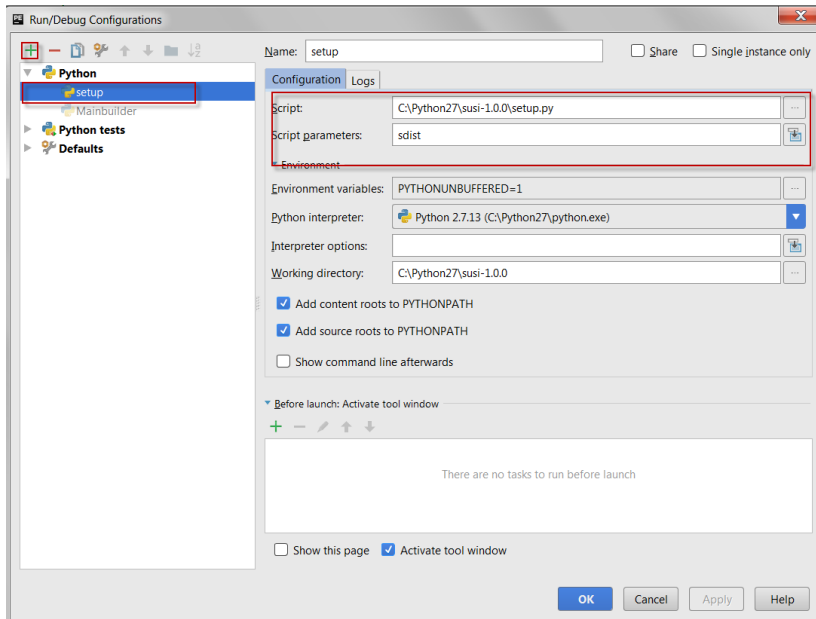
Taula A3.1.1. Codi python

A3.2 Afegir paràmetres sdist al setup

Des del fitxer setup.py amb el botó dret sobre el “Run” i es selecciona l'opció de “add parameters” per afegir el paràmetres “sdist”



Imatge A3.2.1. Paràmetres



Imatge A3.2.2. Configuració

A3.3 Executar setup.py

Crearà el directori dist. Dintre d'aquest directori crearà un fitxer comprimit amb les classes emmagatzemades dintre del directori susi .

Nom del fitxer = name + versió + .zip

Exemple per la versió 1.0.0 del SuSi: susi-1.0.0.zip

A3.4 Com instal·lar aquestes classes.

Cal descomprimir el fitxer, que crearà una sèrie de directoris, fer servir la comanda: *"setup.py install"*; i seguir les indicacions del manual

Annex A4 – Fitxers utilitzats

Degut a la grandària dels fitxers no es poden annexar en aquest document sinó que s'han emmagatzemat en el CD aportat juntament amb la memòria i el instal·lable del programa segons la següent estructura:

A4.1 Fitxers de càrrega

En aquesta carpeta s'han emmagatzemat els fitxers en format "txt" utilitzats amb les especificacions dels sistemes que s'han utilitzat per realitzar els casos de prova presentats en el document.

A4.2 Fitxers d'entrada

En aquesta carpeta s'han emmagatzemat els fitxers de monòmers en format PREPI que s'han utilitzat per realitzar els casos de prova presentats en el document.

A4.3 Fitxers sortida

En aquesta carpeta s'han emmagatzemat els fitxers amb els sistemes creats en format PDB que s'han obtingut de cada un dels casos de prova presentats en el document.

A4.3.1 PDB coordenades periòdiques

S'han adjuntat els fitxers PDB creats amb el programa dins de la caixa periòdica

A4.3.2 PDB sense coordenades periòdiques

S'han adjuntat els fitxers PDB creats amb el programa sense tenir en compte les dimensions de la caixa per a la seva posició però sí per a calcular les col·lisions.

A4.3.3 Taula evolució densitat

S'adjunta la taula amb els resultats de l'evolució de la densitat en funció del temps dels tres sistemes:

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
0	0,9149	0	0,4622	10	0,4585
10	0,9187	20	0,4919	20	0,4762
20	0,9258	40	0,5245	30	0,4938

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
30	0,9219	60	0,5624	40	0,5133
40	0,9226	80	0,6014	50	0,5472
50	0,9229	100	0,6392	60	0,5663
60	0,9232	120	0,6822	70	0,5893
70	0,923	140	0,7188	80	0,6079
80	0,9234	160	0,733	90	0,6282
90	0,9236	180	0,7458	100	0,6508
100	0,9253	200	0,7584	110	0,6731
110	0,9252	220	0,7642	120	0,6953
120	0,9278	240	0,77	130	0,7149
130	0,9288	260	0,777	140	0,7393
140	0,9282	280	0,7814	150	0,7638
150	0,9289	300	0,7827	160	0,7814
160	0,9276	320	0,7848	170	0,8025
170	0,9284	340	0,786	180	0,8219
180	0,9324	360	0,7886	190	0,8457
190	0,9314	380	0,7938	200	0,8674
200	0,9335	400	0,7921	210	0,8779
210	0,9318	420	0,7967	220	0,8877
220	0,9334	440	0,795	230	0,9036
230	0,9338	460	0,798	240	0,9116
240	0,9332	480	0,8013	250	0,9243
250	0,9334	500	0,8032	260	0,9364
260	0,9323	520	0,8059	270	0,9446
270	0,9336	540	0,8081	280	0,9584
280	0,9349	560	0,8108	290	0,9653
290	0,9332	580	0,8126	300	0,9765
300	0,9335	600	0,8171	310	0,9849
310	0,9321	620	0,814	320	0,9935
320	0,9323	640	0,8187	330	0,9999
330	0,9347	660	0,8165	340	1,0072
340	0,9352	680	0,8221	350	1,0141
350	0,9352	700	0,8209	360	1,0202
360	0,9379	720	0,8216	370	1,0204
370	0,9379	740	0,8246	380	1,0256
380	0,9393	760	0,8247	390	1,0281
390	0,9374	780	0,8258	400	1,0288
400	0,9384	800	0,8286	410	1,0301
410	0,9397	820	0,8293	420	1,029
420	0,9409	840	0,8315	430	1,0319
430	0,9424	860	0,8333	440	1,0325

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
440	0,9402	880	0,8359	450	1,0328
450	0,9408	900	0,8341	460	1,034
460	0,9409	920	0,8383	470	1,0369
470	0,9402	940	0,8416	480	1,042
480	0,9417	960	0,838	490	1,0401
490	0,9392	980	0,8416	500	1,04
500	0,9413	1000	0,8404	510	1,0402
510	0,9389	1020	0,8423	520	1,0417
520	0,9397	1040	0,8416	530	1,0435
530	0,938	1060	0,8405	540	1,0436
540	0,939	1080	0,8438	550	1,0456
550	0,94	1100	0,8436	560	1,0459
560	0,9397	1120	0,8453	570	1,0453
570	0,9419	1140	0,846	580	1,0468
580	0,9417	1160	0,8475	590	1,0464
590	0,9428	1180	0,8493	600	1,0486
600	0,9428	1200	0,8505	610	1,0496
610	0,9431	1220	0,8518	620	1,0537
620	0,9426	1240	0,8549	630	1,058
630	0,9434	1260	0,8559	640	1,0626
640	0,9424	1280	0,8592	650	1,0595
650	0,943	1300	0,8577	660	1,0637
660	0,9427	1320	0,8576	670	1,0663
670	0,9419	1340	0,8552	680	1,0676
680	0,9415	1360	0,8577	690	1,0674
690	0,9401	1380	0,8592	700	1,0714
700	0,9421	1400	0,8604	710	1,0709
710	0,9424	1420	0,8589	720	1,07
720	0,9421	1440	0,8581	730	1,0726
730	0,9417	1460	0,8597	740	1,0748
740	0,9422	1480	0,8618	750	1,0717
750	0,9418	1500	0,8608	760	1,0729
760	0,943	1520	0,8607	770	1,0713
770	0,943	1540	0,8609	780	1,0723
780	0,943	1560	0,8592	790	1,0759
790	0,9441	1580	0,8609	800	1,0712
800	0,9433	1600	0,86	810	1,0726
810	0,9435	1620	0,8606	820	1,0688
820	0,9424	1640	0,8622	830	1,0733
830	0,9435	1660	0,8625	840	1,075
840	0,9424	1680	0,8624	850	1,0721

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
850	0,9438	1700	0,8647	860	1,0752
860	0,9458	1720	0,8635	870	1,0736
870	0,9434	1740	0,8633	880	1,0746
880	0,943	1760	0,8618	890	1,0716
890	0,944	1780	0,8634	900	1,0762
900	0,9438	1800	0,8613	910	1,0746
910	0,9429	1820	0,8649	920	1,077
920	0,9437	1840	0,8665	930	1,0779
930	0,9435	1860	0,8658	940	1,0768
940	0,942	1880	0,8658	950	1,0761
950	0,9417	1900	0,8652	960	1,0777
960	0,9431	1920	0,864	970	1,0792
970	0,9439	1940	0,8643	980	1,0798
980	0,9428	1960	0,867	990	1,0782
990	0,9422	1980	0,8668	1000	1,0806
1000	0,9434	2000	0,8655	1010	1,0801
1010	0,942	2020	0,8645	1020	1,081
1020	0,9429	2040	0,8649	1030	1,0823
1030	0,943	2060	0,8653	1040	1,0806
1040	0,9418	2080	0,864	1050	1,0791
1050	0,9416	2100	0,8626	1060	1,0791
1060	0,9431	2120	0,8654	1070	1,0802
1070	0,9425	2140	0,8657	1080	1,0826
1080	0,9423	2160	0,8653	1090	1,0823
1090	0,9436	2180	0,8643	1100	1,082
1100	0,9427	2200	0,8642	1110	1,0818
1110	0,9442	2220	0,8658	1120	1,0843
1120	0,9434	2240	0,8648	1130	1,0841
1130	0,9439	2260	0,8644	1140	1,085
1140	0,9435	2280	0,8669	1150	1,0847
1150	0,9442	2300	0,8659	1160	1,0854
1160	0,9443	2320	0,8658	1170	1,0841
1170	0,9434	2340	0,8661	1180	1,0844
1180	0,944	2360	0,8649	1190	1,0838
1190	0,9442	2380	0,868	1200	1,0874
1200	0,9447	2400	0,87	1210	1,0897
1210	0,9443	2420	0,8691	1220	1,0902
1220	0,9452	2440	0,8697	1230	1,0894
1230	0,9437	2460	0,8691	1240	1,0906
1240	0,9437	2480	0,8691	1250	1,0904
1250	0,9442	2500	0,8669	1260	1,0924

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
1260	0,9449	2520	0,868	1270	1,0922
1270	0,9436	2540	0,8685	1280	1,0917
1280	0,9429	2560	0,8688	1290	1,0931
1290	0,9431	2580	0,8685	1300	1,0937
1300	0,9444	2600	0,87	1310	1,0941
1310	0,9437	2620	0,8694	1320	1,093
1320	0,9435	2640	0,8699	1330	1,0924
1330	0,9438	2660	0,8694	1340	1,0962
1340	0,9448	2680	0,8712	1350	1,0941
1350	0,944	2700	0,8698	1360	1,0923
1360	0,9442	2720	0,8684	1370	1,0915
1370	0,9449	2740	0,8698	1380	1,0929
1380	0,9445	2760	0,8699	1390	1,0923
1390	0,9444	2780	0,8698	1400	1,0926
1400	0,9442	2800	0,8695	1410	1,0893
1410	0,9434	2820	0,8683	1420	1,0937
1420	0,9437	2840	0,8699	1430	1,0933
1430	0,9446	2860	0,8689	1440	1,091
1440	0,9437	2880	0,8715	1450	1,0889
1450	0,9449	2900	0,8709	1460	1,0921
1460	0,9439	2920	0,8708	1470	1,0924
1470	0,9434	2940	0,8703	1480	1,0914
1480	0,9458	2960	0,8703	1490	1,0911
1490	0,9446	2980	0,8718	1500	1,0923
1500	0,945	3000	0,8713	1510	1,0927
1510	0,9437	3020	0,8724	1520	1,0936
1520	0,9442	3040	0,8714	1530	1,0937
1530	0,9435	3060	0,8707	1540	1,093
1540	0,9446	3080	0,8719	1550	1,0947
1550	0,9448	3100	0,8722	1560	1,0954
1560	0,9429	3120	0,8718	1570	1,0914
1570	0,9441	3140	0,8712	1580	1,0961
1580	0,9435	3160	0,8721	1590	1,0933
1590	0,9442	3180	0,8713	1600	1,0936
1600	0,9456	3200	0,8716	1610	1,0933
1610	0,9454	3220	0,8724	1620	1,0968
1620	0,9446	3240	0,8714	1630	1,0935
1630	0,9461	3260	0,8742	1640	1,0956
1640	0,9458	3280	0,8726	1650	1,0954
1650	0,9445	3300	0,8723	1660	1,0936
1660	0,9461	3320	0,8716	1670	1,0953

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
1670	0,9456	3340	0,8732	1680	1,0947
1680	0,945	3360	0,8721	1690	1,0967
1690	0,9461	3380	0,8732	1700	1,0949
1700	0,9454	3400	0,8736	1710	1,0968
1710	0,9449	3420	0,8708	1720	1,0975
1720	0,9448	3440	0,8742	1730	1,0958
1730	0,9467	3460	0,8721	1740	1,096
1740	0,9465	3480	0,873	1750	1,0948
1750	0,9453	3500	0,8738	1760	1,0958
1760	0,9451	3520	0,875	1770	1,0961
1770	0,945	3540	0,8726	1780	1,0976
1780	0,9445	3560	0,8728	1790	1,0982
1790	0,9454	3580	0,8724	1800	1,0987
1800	0,9445	3600	0,8707	1810	1,0968
1810	0,9446	3620	0,8734	1820	1,0964
1820	0,9448	3640	0,8716	1830	1,1009
1830	0,9461	3660	0,8728	1840	1,1008
1840	0,945	3680	0,8726	1850	1,0994
1850	0,9452	3700	0,8746	1860	1,0987
1860	0,9447	3720	0,8733	1870	1,1006
1870	0,9444	3740	0,8748	1880	1,0994
1880	0,9436	3760	0,8719	1890	1,1009
1890	0,9449	3780	0,8739	1900	1,1006
1900	0,9453	3800	0,8747	1910	1,1008
1910	0,9466	3820	0,8731	1920	1,0983
1920	0,9479	3840	0,8715	1930	1,1016
1930	0,9453	3860	0,8731	1940	1,1002
1940	0,9464	3880	0,872	1950	1,099
1950	0,946	3900	0,8741	1960	1,0989
1960	0,945	3920	0,8749	1970	1,0995
1970	0,9456	3940	0,8718	1980	1,1012
1980	0,9456	3960	0,8738	1990	1,0996
1990	0,9453	3980	0,8729	2000	1,1003
2000	0,9454	4000	0,8743	2010	1,1018
2010	0,9452	4020	0,8735	2020	1,1038
2020	0,9458	4040	0,8746	2030	1,1036
2030	0,9455	4060	0,874	2040	1,1022
2040	0,9447	4080	0,8734	2050	1,1013
2050	0,9449	4100	0,8717	2060	1,101
2060	0,9455	4120	0,8733	2070	1,1025
2070	0,9459	4140	0,8764	2080	1,1047

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
2080	0,9452	4160	0,8737	2090	1,1064
2090	0,9458	4180	0,8763	2100	1,1065
2100	0,946	4200	0,8747	2110	1,1068
2110	0,9445	4220	0,8727	2120	1,1057
2120	0,9459	4240	0,873	2130	1,1056
2130	0,9451	4260	0,8759	2140	1,1054
2140	0,9463	4280	0,8737	2150	1,1046
2150	0,9454	4300	0,8745	2160	1,1072
2160	0,9472	4320	0,8732	2170	1,1051
2170	0,9462	4340	0,8735	2180	1,1078
2180	0,947	4360	0,8717	2190	1,1069
2190	0,9478	4380	0,8724	2200	1,1048
2200	0,9462	4400	0,8728	2210	1,105
2210	0,947	4420	0,8734	2220	1,1051
2220	0,9455	4440	0,8735	2230	1,1079
2230	0,946	4460	0,8721	2240	1,1062
2240	0,9463	4480	0,8722	2250	1,1076
2250	0,9473	4500	0,8741	2260	1,1093
2260	0,9461	4520	0,8725	2270	1,1049
2270	0,9463	4540	0,874	2280	1,1053
2280	0,9471	4560	0,8727	2290	1,1046
2290	0,9481	4580	0,8732	2300	1,1046
2300	0,9461	4600	0,8741	2310	1,1085
2310	0,9472	4620	0,8724	2320	1,1055
2320	0,9467	4640	0,8733	2330	1,1063
2330	0,9465	4660	0,8752	2340	1,1065
2340	0,9454	4680	0,8734	2350	1,1085
2350	0,9477	4700	0,8711	2360	1,1098
2360	0,9478	4720	0,8734	2370	1,1047
2370	0,9466	4740	0,8738	2380	1,1048
2380	0,9471	4760	0,8737	2390	1,106
2390	0,9473	4780	0,8733	2400	1,1075
2400	0,9484	4800	0,8728	2410	1,1087
2410	0,9474	4820	0,8735	2420	1,1077
2420	0,9465	4840	0,8716	2430	1,1068
2430	0,9475	4860	0,872	2440	1,1029
2440	0,9472	4880	0,8738	2450	1,1034
2450	0,9467	4900	0,875	2460	1,1046
2460	0,9473	4920	0,8751	2470	1,104
2470	0,9477	4940	0,8721	2480	1,1067
2480	0,9472	4960	0,8734	2490	1,1061

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
2490	0,9469	4980	0,873	2500	1,1068
2500	0,9476	5000	0,873	2510	1,1059
2510	0,9455	5020	0,8731	2520	1,1083
2520	0,9462	5040	0,8758	2530	1,1086
2530	0,9462	5060	0,8755	2540	1,1084
2540	0,9462	5080	0,8746	2550	1,107
2550	0,9463	5100	0,8745	2560	1,1072
2560	0,9477	5120	0,8744	2570	1,1089
2570	0,9481	5140	0,8736	2580	1,1065
2580	0,9467	5160	0,8751	2590	1,1105
2590	0,9473	5180	0,8743	2600	1,1098
2600	0,9455	5200	0,8759	2610	1,1091
2610	0,9455	5220	0,8733	2620	1,1086
2620	0,9442	5240	0,8732	2630	1,1106
2630	0,9468	5260	0,8744	2640	1,1105
2640	0,9471	5280	0,8751	2650	1,111
2650	0,9453	5300	0,874	2660	1,1111
2660	0,9458	5320	0,8732	2670	1,1093
2670	0,9458	5340	0,8749	2680	1,1103
2680	0,9459	5360	0,8736	2690	1,1106
2690	0,946	5380	0,8749	2700	1,1081
2700	0,9469	5400	0,8756	2710	1,1123
2710	0,9452	5420	0,8747	2720	1,1105
2720	0,9454	5440	0,8745	2730	1,1087
2730	0,9458	5460	0,8772	2740	1,1085
2740	0,9467	5480	0,8767	2750	1,1094
2750	0,9456	5500	0,8753	2760	1,1083
2760	0,9448	5520	0,8754	2770	1,1091
2770	0,9464	5540	0,8763	2780	1,1098
2780	0,946	5560	0,8771	2790	1,1099
2790	0,947	5580	0,8747	2800	1,1114
2800	0,9459	5600	0,8762	2810	1,1102
2810	0,9445	5620	0,8754	2820	1,1102
2820	0,9451	5640	0,8757	2830	1,1115
2830	0,9449	5660	0,875	2840	1,1109
2840	0,946	5680	0,8747	2850	1,1086
2850	0,9443	5700	0,8746	2860	1,113
2860	0,946	5720	0,8728	2870	1,1103
2870	0,9439	5740	0,8755	2880	1,112
2880	0,9439	5760	0,8744	2890	1,1098
2890	0,9447	5780	0,876	2900	1,111

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
2900	0,9457	5800	0,8775	2910	1,1116
2910	0,9447	5820	0,8775	2920	1,114
2920	0,9441	5840	0,8777	2930	1,11
2930	0,9447	5860	0,877	2940	1,1092
2940	0,9438	5880	0,8759	2950	1,1107
2950	0,9453	5900	0,8771	2960	1,1128
2960	0,9456	5920	0,8776	2970	1,1144
2970	0,943	5940	0,8775	2980	1,1139
2980	0,9442	5960	0,8765	2990	1,1116
2990	0,9439	5980	0,8775	3000	1,1104
3000	0,9435	6000	0,8758	3010	1,1103
3010	0,9458	6020	0,8769	3020	1,1104
3020	0,9439	6040	0,8774	3030	1,1111
3030	0,9455	6060	0,8784	3040	1,1098
3040	0,9445	6080	0,8774	3050	1,11
3050	0,9446	6100	0,875	3060	1,11
3060	0,9429	6120	0,8758	3070	1,1107
3070	0,9427	6140	0,8758	3080	1,1093
3080	0,9425	6160	0,8776	3090	1,1122
3090	0,9426	6180	0,8767	3100	1,1097
3100	0,9436	6200	0,8765	3110	1,1122
3110	0,9429	6220	0,8776	3120	1,1137
3120	0,944	6240	0,877	3130	1,111
3130	0,9446	6260	0,8756	3140	1,1112
3140	0,9447	6280	0,8764	3150	1,11
3150	0,9447	6300	0,8766	3160	1,1104
3160	0,9449	6320	0,8771	3170	1,1119
3170	0,945	6340	0,8772	3180	1,1142
3180	0,945	6360	0,8755	3190	1,1135
3190	0,9437	6380	0,8797	3200	1,1132
3200	0,9448	6400	0,8774	3210	1,1117
3210	0,9451	6420	0,8781	3220	1,114
3220	0,9454	6440	0,8782	3230	1,1118
3230	0,9447	6460	0,8796	3240	1,1118
3240	0,9438	6480	0,8786	3250	1,1136
3250	0,9431	6500	0,8775	3260	1,1129
3260	0,9438	6520	0,8765	3270	1,1138
3270	0,9446	6540	0,8774	3280	1,115
3280	0,9454	6560	0,8764	3290	1,1166
3290	0,9456	6580	0,8758	3300	1,1138
3300	0,9442	6600	0,8767	3310	1,1157

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
3310	0,944	6620	0,8771	3320	1,117
3320	0,9436	6640	0,8758	3330	1,1176
3330	0,9449	6660	0,8759	3340	1,1149
3340	0,943	6680	0,8746	3350	1,1134
3350	0,9439	6700	0,8745	3360	1,1141
3360	0,9441	6720	0,8749	3370	1,1153
3370	0,9434	6740	0,8764	3380	1,1162
3380	0,9437	6760	0,8752	3390	1,1161
3390	0,9441	6780	0,8754	3400	1,1138
3400	0,9436	6800	0,8764	3410	1,1168
3410	0,944	6820	0,8769	3420	1,1153
3420	0,9442	6840	0,8775	3430	1,1174
3430	0,9441	6860	0,8774	3440	1,1157
3440	0,942	6880	0,8764	3450	1,1167
3450	0,9437	6900	0,8748	3460	1,1158
3460	0,9438	6920	0,8743	3470	1,1161
3470	0,9449	6940	0,8756	3480	1,1187
3480	0,944	6960	0,8778	3490	1,1151
3490	0,9439	6980	0,8749	3500	1,118
3500	0,9454	7000	0,8756	3510	1,115
3510	0,945	7020	0,8759	3520	1,1162
3520	0,9455	7040	0,8771	3530	1,1201
3530	0,944	7060	0,8749	3540	1,1174
3540	0,9439	7080	0,8764	3550	1,1162
3550	0,9437	7100	0,8766	3560	1,1181
3560	0,9445	7120	0,8755	3570	1,1182
3570	0,9444	7140	0,8777	3580	1,1187
3580	0,9437	7160	0,877	3590	1,1195
3590	0,9445	7180	0,8772	3600	1,1172
3600	0,944	7200	0,8765	3610	1,1189
3610	0,9439	7220	0,8772	3620	1,1184
3620	0,9434	7240	0,8783	3630	1,1174
3630	0,9434	7260	0,8774	3640	1,1197
3640	0,9452	7280	0,879		
3650	0,9444	7300	0,8782		
3660	0,9447	7320	0,8774		
3670	0,9435	7340	0,877		
3680	0,9447	7360	0,8768		
3690	0,9443	7380	0,8772		
3700	0,9446	7400	0,8764		
3710	0,9432	7420	0,8772		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
3720	0,9441	7440	0,8754		
3730	0,9444	7460	0,8769		
3740	0,9449	7480	0,8768		
3750	0,9442	7500	0,8761		
3760	0,945	7520	0,8772		
3770	0,9428	7540	0,8768		
3780	0,9442	7560	0,8786		
3790	0,944	7580	0,8777		
3800	0,9439	7600	0,8779		
3810	0,9434	7620	0,8789		
3820	0,9424	7640	0,8792		
3830	0,9451	7660	0,8787		
3840	0,9458	7680	0,8779		
3850	0,9448	7700	0,8775		
3860	0,945	7720	0,8767		
3870	0,945	7740	0,8761		
3880	0,9444	7760	0,8783		
3890	0,9446	7780	0,8764		
3900	0,943	7800	0,8778		
3910	0,9442	7820	0,8768		
3920	0,944	7840	0,8778		
3930	0,9436	7860	0,8774		
3940	0,9431	7880	0,878		
3950	0,9441	7900	0,8777		
3960	0,9427	7920	0,8792		
3970	0,9417	7940	0,8779		
3980	0,9431	7960	0,8773		
3990	0,9425	7980	0,878		
4000	0,944	8000	0,877		
4010	0,9425	8020	0,8773		
4020	0,9424	8040	0,8793		
4030	0,9426	8060	0,8782		
4040	0,944	8080	0,878		
4050	0,9444	8100	0,8773		
4060	0,943	8120	0,8774		
4070	0,9438	8140	0,8765		
4080	0,9425	8160	0,878		
4090	0,9434	8180	0,8784		
4100	0,9429	8200	0,8765		
4110	0,9426	8220	0,8769		
4120	0,9429	8240	0,8772		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
4130	0,9422	8260	0,8771		
4140	0,9419	8280	0,8785		
4150	0,9441	8300	0,8774		
4160	0,9429	8320	0,8795		
4170	0,9426	8340	0,8779		
4180	0,9437	8360	0,8773		
4190	0,9434	8380	0,8767		
4200	0,9432	8400	0,8759		
4210	0,9438	8420	0,876		
4220	0,9431	8440	0,8768		
4230	0,9429	8460	0,8766		
4240	0,9423	8480	0,8766		
4250	0,9416	8500	0,8765		
4260	0,9424	8520	0,8773		
4270	0,9432	8540	0,8748		
4280	0,943	8560	0,8775		
4290	0,9431	8580	0,8767		
4300	0,9427	8600	0,8762		
4310	0,9428	8620	0,8759		
4320	0,9416	8640	0,8753		
4330	0,943	8660	0,8736		
4340	0,9426	8680	0,8753		
4350	0,9424	8700	0,8765		
4360	0,9439	8720	0,8766		
4370	0,943	8740	0,8755		
4380	0,9442	8760	0,8767		
4390	0,9452	8780	0,8773		
4400	0,9451	8800	0,877		
4410	0,9436	8820	0,8751		
4420	0,9446	8840	0,8771		
4430	0,9452	8860	0,878		
4440	0,9458	8880	0,8791		
4450	0,9459	8900	0,8769		
4460	0,9449	8920	0,8781		
4470	0,9452	8940	0,8779		
4480	0,9448	8960	0,8771		
4490	0,9453	8980	0,8768		
4500	0,9438	9000	0,8778		
4510	0,9448	9020	0,8774		
4520	0,9439	9040	0,8771		
4530	0,9437	9060	0,877		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
4540	0,9436	9080	0,877		
4550	0,9428	9100	0,8779		
4560	0,9432	9120	0,8776		
4570	0,9441	9140	0,8783		
4580	0,9436	9160	0,8775		
4590	0,9443	9180	0,8776		
4600	0,9439	9200	0,8806		
4610	0,9456	9220	0,8777		
4620	0,9454	9240	0,8775		
4630	0,9438	9260	0,8784		
4640	0,9428	9280	0,8776		
4650	0,9445	9300	0,8789		
4660	0,9441	9320	0,8783		
4670	0,9437	9340	0,8781		
4680	0,9446	9360	0,8784		
4690	0,9434	9380	0,878		
4700	0,943	9400	0,8786		
4710	0,9433	9420	0,8785		
4720	0,9439	9440	0,8773		
4730	0,9436	9460	0,8775		
4740	0,9449	9480	0,8782		
4750	0,9458	9500	0,879		
4760	0,9457	9520	0,8773		
4770	0,9458	9540	0,8792		
4780	0,9456	9560	0,877		
4790	0,9451	9580	0,8772		
4800	0,9445	9600	0,878		
4810	0,9447	9620	0,8771		
4820	0,9443	9640	0,8781		
4830	0,9439	9660	0,8774		
4840	0,9452	9680	0,8769		
4850	0,9455	9700	0,8768		
4860	0,9452	9720	0,8761		
4870	0,9445	9740	0,8772		
4880	0,9442	9760	0,876		
4890	0,9433	9780	0,876		
4900	0,9443	9800	0,8767		
4910	0,9445	9820	0,8758		
4920	0,9442	9840	0,8766		
4930	0,9437	9860	0,8779		
4940	0,9451	9880	0,8757		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
4950	0,9449	9900	0,8748		
4960	0,9427	9920	0,8765		
4970	0,943	9940	0,8782		
4980	0,9453	9960	0,8777		
4990	0,9442	9980	0,8775		
5000	0,9446	10000	0,8764		
5010	0,9437	10020	0,8775		
5020	0,9436	10040	0,8762		
5030	0,9441	10060	0,8777		
5040	0,9454	10080	0,8772		
5050	0,944	10100	0,8771		
5060	0,9443	10120	0,876		
5070	0,9448	10140	0,8782		
5080	0,9445	10160	0,8778		
5090	0,9441	10180	0,878		
5100	0,9434	10200	0,8773		
5110	0,9432	10220	0,8769		
5120	0,9439	10240	0,8782		
5130	0,946	10260	0,8778		
5140	0,945	10280	0,878		
5150	0,9455	10300	0,8797		
5160	0,9427	10320	0,8781		
5170	0,9445	10340	0,8795		
5180	0,9449	10360	0,8783		
5190	0,9448	10380	0,8791		
5200	0,9452	10400	0,88		
5210	0,9443	10420	0,8787		
5220	0,947	10440	0,8785		
5230	0,9465	10460	0,8781		
5240	0,9455	10480	0,8807		
5250	0,9461	10500	0,8791		
5260	0,9451	10520	0,8794		
5270	0,9438	10540	0,8815		
5280	0,9442	10560	0,8802		
5290	0,9446	10580	0,8814		
5300	0,9453	10600	0,8814		
5310	0,946	10620	0,8797		
5320	0,9443	10640	0,8812		
5330	0,9459	10660	0,8801		
5340	0,9452	10680	0,8788		
5350	0,9454	10700	0,8789		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
5360	0,946	10720	0,8794		
5370	0,9448	10740	0,8807		
5380	0,9435	10760	0,8798		
5390	0,9456	10780	0,8794		
5400	0,9453	10800	0,8799		
5410	0,9443	10820	0,8802		
5420	0,9453	10840	0,8785		
5430	0,9464	10860	0,8811		
5440	0,9459	10880	0,8807		
5450	0,9455	10900	0,8797		
5460	0,9458	10920	0,8798		
5470	0,9453	10940	0,8803		
5480	0,9464	10960	0,8808		
5490	0,9466	10980	0,8806		
5500	0,946	11000	0,8804		
5510	0,9468	11020	0,881		
5520	0,9456	11040	0,881		
5530	0,9458	11060	0,8833		
5540	0,9472	11080	0,88		
5550	0,947	11100	0,8817		
5560	0,9456	11120	0,8803		
5570	0,9454	11140	0,8809		
5580	0,9463	11160	0,8808		
5590	0,9463	11180	0,881		
5600	0,9467	11200	0,8803		
5610	0,9467	11220	0,8808		
5620	0,945	11240	0,8814		
5630	0,9453	11260	0,8799		
5640	0,9472	11280	0,881		
5650	0,9469	11300	0,8824		
5660	0,9457	11320	0,8817		
5670	0,9461	11340	0,8814		
5680	0,9447	11360	0,8827		
5690	0,9463	11380	0,8821		
5700	0,9479	11400	0,8826		
5710	0,9471	11420	0,8816		
5720	0,946	11440	0,8819		
5730	0,9466	11460	0,8793		
5740	0,9462	11480	0,8813		
5750	0,9456	11500	0,882		
5760	0,9466	11520	0,8842		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
5770	0,9463	11540	0,883		
5780	0,9474	11560	0,8825		
5790	0,9473	11580	0,8815		
5800	0,9469	11600	0,8813		
5810	0,9467	11620	0,8823		
5820	0,9466	11640	0,8813		
5830	0,9464	11660	0,8812		
5840	0,9469	11680	0,8807		
5850	0,9462	11700	0,8804		
5860	0,946	11720	0,8807		
5870	0,9461	11740	0,8807		
5880	0,9469	11760	0,8806		
5890	0,9461	11780	0,8834		
5900	0,9462	11800	0,884		
5910	0,9456	11820	0,8831		
5920	0,9454	11840	0,8814		
5930	0,9452	11860	0,8814		
5940	0,9449	11880	0,8826		
5950	0,9458	11900	0,8832		
5960	0,9461	11920	0,8816		
5970	0,9469	11940	0,8833		
5980	0,9478	11960	0,8818		
5990	0,9476	11980	0,8826		
6000	0,9475	12000	0,8832		
6010	0,9474	12020	0,8811		
6020	0,9481	12040	0,881		
6030	0,9469	12060	0,8792		
6040	0,9456	12080	0,8804		
6050	0,9462	12100	0,8793		
6060	0,9467	12120	0,8811		
6070	0,9471	12140	0,8807		
6080	0,9469	12160	0,88		
6090	0,9485	12180	0,8812		
6100	0,9472	12200	0,881		
6110	0,9479	12220	0,8811		
6120	0,9478	12240	0,8812		
6130	0,9477	12260	0,8809		
6140	0,9472	12280	0,8813		
6150	0,9484	12300	0,8817		
6160	0,9471	12320	0,8811		
6170	0,9494	12340	0,8827		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
6180	0,9485	12360	0,883		
6190	0,9475	12380	0,8823		
6200	0,9479	12400	0,8806		
6210	0,9471	12420	0,879		
6220	0,9473	12440	0,881		
6230	0,9477	12460	0,8814		
6240	0,9475	12480	0,8805		
6250	0,9476	12500	0,8814		
6260	0,9462	12520	0,8818		
6270	0,9466	12540	0,8815		
6280	0,9474	12560	0,8829		
6290	0,9455	12580	0,8815		
6300	0,947	12600	0,8826		
6310	0,9463	12620	0,8835		
6320	0,947	12640	0,8824		
6330	0,9477	12660	0,8822		
6340	0,9472	12680	0,8824		
6350	0,9482	12700	0,8827		
6360	0,9475	12720	0,8836		
6370	0,9476	12740	0,882		
6380	0,9482	12760	0,8835		
6390	0,9471	12780	0,8834		
6400	0,9469	12800	0,8826		
6410	0,9473	12820	0,8837		
6420	0,9481	12840	0,8825		
6430	0,9481	12860	0,8812		
6440	0,9476	12880	0,8812		
6450	0,9472	12900	0,8823		
6460	0,9468	12920	0,8838		
6470	0,9475	12940	0,8832		
6480	0,9478	12960	0,8826		
6490	0,9473	12980	0,8816		
6500	0,9467	13000	0,8838		
6510	0,9474	13020	0,8821		
6520	0,9465	13040	0,8825		
6530	0,9465	13060	0,8814		
6540	0,9488	13080	0,8813		
6550	0,9475	13100	0,8822		
6560	0,9472	13120	0,8825		
6570	0,9468	13140	0,8819		
6580	0,9469	13160	0,8828		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
6590	0,9478	13180	0,8833		
6600	0,9472	13200	0,8815		
6610	0,9474	13220	0,8815		
6620	0,948	13240	0,8828		
6630	0,948	13260	0,882		
6640	0,9484	13280	0,8823		
6650	0,9472	13300	0,8822		
6660	0,947	13320	0,8811		
6670	0,9482	13340	0,8798		
6680	0,9473	13360	0,8791		
6690	0,947	13380	0,8804		
6700	0,9473	13400	0,8801		
6710	0,9477	13420	0,8808		
6720	0,9481	13440	0,8807		
6730	0,9477	13460	0,8821		
6740	0,9478	13480	0,8819		
6750	0,9472	13500	0,8829		
6760	0,9486	13520	0,8843		
6770	0,9499	13540	0,8814		
6780	0,9489	13560	0,8837		
6790	0,9482	13580	0,8831		
6800	0,9489	13600	0,8829		
6810	0,9488	13620	0,8816		
6820	0,9472	13640	0,8826		
6830	0,949	13660	0,8814		
6840	0,9488	13680	0,881		
6850	0,9491	13700	0,8828		
6860	0,9484	13720	0,8827		
6870	0,9506	13740	0,8805		
6880	0,9487	13760	0,8838		
6890	0,9491	13780	0,8815		
6900	0,95	13800	0,8833		
6910	0,9482	13820	0,8836		
6920	0,9485	13840	0,8829		
6930	0,9488	13860	0,8822		
6940	0,9477	13880	0,8818		
6950	0,949	13900	0,881		
6960	0,9486	13920	0,882		
6970	0,9478	13940	0,8818		
6980	0,9476	13960	0,8829		
6990	0,9466	13980	0,8827		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
7000	0,9473	14000	0,8825		
7010	0,9483	14020	0,8834		
7020	0,9479	14040	0,8835		
7030	0,9468	14060	0,8826		
7040	0,9467	14080	0,8824		
7050	0,9487	14100	0,8829		
7060	0,9476	14120	0,8844		
7070	0,9473	14140	0,8833		
7080	0,9471	14160	0,8837		
7090	0,9466	14180	0,8835		
7100	0,9471	14200	0,8825		
7110	0,9485	14220	0,8844		
7120	0,9477	14240	0,8817		
7130	0,9479	14260	0,8856		
7140	0,947	14280	0,8847		
7150	0,9471	14300	0,8833		
7160	0,947	14320	0,883		
7170	0,9496	14340	0,884		
7180	0,9483	14360	0,8853		
7190	0,9489	14380	0,8845		
7200	0,9476	14400	0,8843		
7210	0,9483	14420	0,8853		
7220	0,9492	14440	0,8817		
7230	0,9498	14460	0,8835		
7240	0,95	14480	0,8845		
7250	0,9498	14500	0,8824		
7260	0,9475	14520	0,8837		
7270	0,9491	14540	0,8838		
7280	0,9501	14560	0,8846		
7290	0,9488	14580	0,883		
7300	0,948	14600	0,8827		
7310	0,9484	14620	0,8825		
7320	0,9477	14640	0,8829		
7330	0,948	14660	0,8819		
7340	0,9493	14680	0,8831		
7350	0,9483	14700	0,8823		
7360	0,9492	14720	0,8834		
7370	0,9477	14740	0,8827		
7380	0,9475	14760	0,883		
7390	0,9494	14780	0,8838		
7400	0,9495	14800	0,8839		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
7410	0,9479	14820	0,8832		
7420	0,9472	14840	0,8833		
7430	0,946	14860	0,8855		
7440	0,9476	14880	0,8843		
7450	0,9477	14900	0,8851		
7460	0,9491	14920	0,8833		
7470	0,9469	14940	0,8828		
7480	0,9474	14960	0,8819		
7490	0,9475	14980	0,8811		
7500	0,9493	15000	0,8833		
7510	0,9482	15020	0,8823		
7520	0,9486	15040	0,8816		
7530	0,9492	15060	0,884		
7540	0,9479	15080	0,8857		
7550	0,9495	15100	0,8849		
7560	0,9481	15120	0,8841		
7570	0,9487	15140	0,8824		
7580	0,9492	15160	0,8833		
7590	0,949	15180	0,881		
7600	0,9499	15200	0,883		
7610	0,9497	15220	0,8831		
7620	0,9491	15240	0,8808		
7630	0,9482	15260	0,8833		
7640	0,9487	15280	0,8822		
7650	0,9491	15300	0,8823		
7660	0,9473	15320	0,8833		
7670	0,9484	15340	0,8846		
7680	0,9493	15360	0,8825		
7690	0,9492	15380	0,8823		
7700	0,9481	15400	0,8834		
7710	0,9468	15420	0,8827		
7720	0,9485	15440	0,8837		
7730	0,9482	15460	0,8823		
7740	0,9483	15480	0,8827		
7750	0,9474	15500	0,883		
7760	0,9474	15520	0,8834		
7770	0,9488	15540	0,8839		
7780	0,9491	15560	0,8831		
7790	0,9483	15580	0,8833		
7800	0,9485	15600	0,8858		
7810	0,947	15620	0,8811		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
7820	0,9469	15640	0,8835		
7830	0,9486	15660	0,881		
7840	0,9483	15680	0,8816		
7850	0,9483	15700	0,8815		
7860	0,9496	15720	0,8835		
7870	0,9492	15740	0,8838		
7880	0,9487	15760	0,8846		
7890	0,9474	15780	0,8839		
7900	0,948	15800	0,8834		
7910	0,9492	15820	0,8828		
7920	0,9489	15840	0,8843		
7930	0,9492	15860	0,8838		
7940	0,949	15880	0,8863		
7950	0,9486	15900	0,8866		
7960	0,9487	15920	0,8855		
7970	0,9485	15940	0,8853		
7980	0,9484	15960	0,8856		
7990	0,9488	15980	0,8866		
8000	0,9494	16000	0,883		
8010	0,9489	16020	0,8849		
8020	0,9487	16040	0,8846		
8030	0,9486	16060	0,8855		
8040	0,9477	16080	0,8817		
8050	0,9501	16100	0,8851		
8060	0,9505	16120	0,8838		
8070	0,9482	16140	0,8829		
8080	0,9486	16160	0,8837		
8090	0,9502	16180	0,8852		
8100	0,9497	16200	0,8861		
8110	0,9495	16220	0,8852		
		16240	0,8841		
		16260	0,8835		
		16280	0,8847		
		16300	0,8843		
		16320	0,8854		
		16340	0,8838		
		16360	0,8856		
		16380	0,8864		
		16400	0,8853		
		16420	0,8842		
		16440	0,884		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
		16460	0,8854		
		16480	0,8864		
		16500	0,885		
		16520	0,8852		
		16540	0,8841		
		16560	0,8857		
		16580	0,8846		
		16600	0,885		
		16620	0,8848		
		16640	0,8846		
		16660	0,8823		
		16680	0,8851		
		16700	0,8837		
		16720	0,8828		
		16740	0,8845		
		16760	0,884		
		16780	0,8859		
		16800	0,8868		
		16820	0,8856		
		16840	0,8851		
		16860	0,8858		
		16880	0,8854		
		16900	0,8842		
		16920	0,8847		
		16940	0,8826		
		16960	0,8842		
		16980	0,8842		
		17000	0,8846		
		17020	0,8839		
		17040	0,8845		
		17060	0,8843		
		17080	0,8827		
		17100	0,8855		
		17120	0,8843		
		17140	0,8824		
		17160	0,884		
		17180	0,8828		
		17200	0,8836		
		17220	0,8831		
		17240	0,8832		
		17260	0,8835		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
		17280	0,8834		
		17300	0,8846		
		17320	0,8838		
		17340	0,8842		
		17360	0,8844		
		17380	0,8842		
		17400	0,8837		
		17420	0,8839		
		17440	0,8832		
		17460	0,8842		
		17480	0,8859		
		17500	0,885		
		17520	0,8841		
		17540	0,8846		
		17560	0,8826		
		17580	0,8839		
		17600	0,8843		
		17620	0,8844		
		17640	0,8859		
		17660	0,8847		
		17680	0,8856		
		17700	0,8862		
		17720	0,8867		
		17740	0,8839		
		17760	0,8833		
		17780	0,8843		
		17800	0,8849		
		17820	0,8858		
		17840	0,8847		
		17860	0,8837		
		17880	0,8852		
		17900	0,8854		
		17920	0,8841		
		17940	0,8858		
		17960	0,8848		
		17980	0,8859		
		18000	0,8842		
		18020	0,8852		
		18040	0,8824		
		18060	0,8833		
		18080	0,8838		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
		18100	0,8847		
		18120	0,8856		
		18140	0,8858		
		18160	0,886		
		18180	0,884		
		18200	0,883		
		18220	0,8835		
		18240	0,8823		
		18260	0,8836		
		18280	0,8849		
		18300	0,8848		
		18320	0,8851		
		18340	0,883		
		18360	0,883		
		18380	0,8816		
		18400	0,8828		
		18420	0,8825		
		18440	0,8832		
		18460	0,8827		
		18480	0,8822		
		18500	0,8831		
		18520	0,8824		
		18540	0,8825		
		18560	0,8835		
		18580	0,8846		
		18600	0,8843		
		18620	0,8846		
		18640	0,8855		
		18660	0,8837		
		18680	0,8829		
		18700	0,8836		
		18720	0,8819		
		18740	0,8841		
		18760	0,8834		
		18780	0,8833		
		18800	0,8841		
		18820	0,8826		
		18840	0,8825		
		18860	0,8838		
		18880	0,8861		
		18900	0,8824		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
		18920	0,8843		
		18940	0,8855		
		18960	0,8855		
		18980	0,8852		
		19000	0,8855		
		19020	0,8841		
		19040	0,8835		
		19060	0,883		
		19080	0,8833		
		19100	0,8856		
		19120	0,8854		
		19140	0,8837		
		19160	0,8845		
		19180	0,8827		
		19200	0,8868		
		19220	0,8848		
		19240	0,8871		
		19260	0,8836		
		19280	0,8835		
		19300	0,8833		
		19320	0,885		
		19340	0,8843		
		19360	0,8839		
		19380	0,8867		
		19400	0,885		
		19420	0,8829		
		19440	0,8862		
		19460	0,8856		
		19480	0,8866		
		19500	0,8856		
		19520	0,8831		
		19540	0,8836		
		19560	0,8852		
		19580	0,8849		
		19600	0,8854		
		19620	0,886		
		19640	0,8848		
		19660	0,8867		
		19680	0,8856		
		19700	0,8862		
		19720	0,8861		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
		19740	0,8835		
		19760	0,8864		
		19780	0,8854		
		19800	0,8845		
		19820	0,8862		
		19840	0,8846		
		19860	0,8848		
		19880	0,8862		
		19900	0,8833		
		19920	0,8851		
		19940	0,8841		
		19960	0,8842		
		19980	0,8858		
		20000	0,885		
		20020	0,8853		
		20040	0,8874		
		20060	0,8861		
		20080	0,8855		
		20100	0,886		
		20120	0,8865		
		20140	0,8867		
		20160	0,8861		
		20180	0,8878		
		20200	0,887		
		20220	0,8868		
		20240	0,8857		
		20260	0,8874		
		20280	0,8855		
		20300	0,8863		
		20320	0,8864		
		20340	0,8886		
		20360	0,8885		
		20380	0,8874		
		20400	0,8885		
		20420	0,8858		
		20440	0,8872		
		20460	0,8875		
		20480	0,8875		
		20500	0,8886		
		20520	0,887		
		20540	0,8882		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
		20560	0,889		
		20580	0,8884		
		20600	0,8864		
		20620	0,886		
		20640	0,8871		
		20660	0,8885		
		20680	0,8878		
		20700	0,8889		
		20720	0,8877		
		20740	0,8878		
		20760	0,8877		
		20780	0,8877		
		20800	0,8875		
		20820	0,8875		
		20840	0,8877		
		20860	0,8876		
		20880	0,8886		
		20900	0,8877		
		20920	0,8872		
		20940	0,887		
		20960	0,8877		
		20980	0,8868		
		21000	0,8873		
		21020	0,8869		
		21040	0,8867		
		21060	0,8866		
		21080	0,888		
		21100	0,8898		
		21120	0,8902		
		21140	0,8895		
		21160	0,8896		
		21180	0,8891		
		21200	0,8891		
		21220	0,8893		
		21240	0,8894		
		21260	0,8897		
		21280	0,8908		
		21300	0,8911		
		21320	0,8904		
		21340	0,8915		
		21360	0,8882		

PS		PE		FPB	
Temps(ps)	densitat PS	temps(ps)	densitat(g/L)	temps(ps)	densitat(g/l)
		21380	0,8881		
		21400	0,8875		
		21420	0,8898		
		21440	0,8878		
		21460	0,8891		
		21480	0,8874		
		21500	0,8871		
		21520	0,8875		
		21540	0,887		
		21560	0,8876		
		21580	0,888		
		21600	0,8889		
		21620	0,8866		
		21640	0,8873		
		21660	0,8879		
		21680	0,8892		
		21700	0,8883		
		21720	0,8886		
		21740	0,8891		
		21760	0,8892		
		21780	0,8895		
		21800	0,8895		
		21820	0,8887		
		21840	0,889		
		21860	0,8889		
		21880	0,8899		
		21900	0,8883		
		21920	0,8902		
		21940	0,8885		
		21960	0,89		
		21980	0,8885		
		22000	0,8908		
		22020	0,8902		
		22040	0,8902		
		22060	0,8892		
		22080	0,8882		
		22100	0,8901		
		22120	0,8898		
		22140	0,8898		

